

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



Trabajo fin de Grado

***Diseño y desarrollo de un sistema web basado en java de
generación de preguntas, revisión entre pares y
competición***

Autor: Grecia Isabel Vera Villajulca

Tutor: Iria Estévez Ayres

Octubre 2015

RESUMEN

En esta memoria de trabajo de fin de grado se va a explicar en detalle el proceso de análisis, diseño e implementación de una aplicación web que tiene fines docentes. Esta aplicación mezcla los conceptos de revisión entre pares y competición educativa, con el fin de aprovechar los beneficios que ofrecen ambas aproximaciones. Por una parte, la revisión entre pares de preguntas servirá para afianzar conceptos teóricos y prácticos de la asignatura, mientras que la competición introduce un elemento lúdico. Con ello se persigue motivar a los estudiantes a estudiar y a profundizar en una materia.

Desde el punto de vista del alumno, el proceso se compone de cuatro fases diferenciadas y obligatorias, correspondiendo las tres primeras a revisión entre pares doble ciego, donde los estudiantes deberán redactar, revisar y corregir preguntas tipo test. Además, deberán puntuar de forma anónima el trabajo hecho por sus propios compañeros. En la última fase deberán responder a las preguntas generadas por sus propios compañeros. Entre el ciclo de revisión entre pares y la fase de competición, el docente tiene la opción de supervisar todo el proceso, puntuando y corrigiendo las preguntas generadas por el alumnado, y, si así lo desea, seleccionando aquellas más adecuadas para la fase de competición. Durante todo el proceso, al finalizar cada fase, los alumnos podrán ver las puntuaciones que les han otorgado tanto sus propios compañeros como el equipo docente.

La aplicación se ha realizado mediante el uso del lenguaje de programación Java siguiendo una estructura modelo vista controlador (MVC). Además hace uso de una base de datos, en este caso MySQL, donde se guardaran todos los datos del juego, desde el momento que se hace el registro de usuario y su contraseña, pasando por todo el progreso del juego, es decir, se guardan las preguntas, revisiones, comentarios, respuestas y puntuaciones que se dan a dicho usuario.

ÍNDICE

RESUMEN	3
INTRODUCCIÓN	15
1.1 MOTIVACIÓN	15
1.1.1 Entorno socio-económico	16
1.2 OBJETIVOS	16
1.3 MARCO REGULADOR.....	17
1.4 ESTRUCTURA DE LA MEMORIA	19
ESTADO DEL ARTE	20
2.1 REVISIÓN ENTRE PARES.....	20
2.2 SISTEMA DE COMPETICIÓN	21
2.3 TECNOLOGÍAS DEL SERVIDOR	21
2.3.1 Sistema Gestor de Base de Datos.....	22
2.3.2 Servidor Web.....	25
2.3.3 Lenguaje de programación.....	26
2.4 TECNOLOGÍAS DEL CLIENTE	27
2.4.1 HTML	28
2.4.2 CSS.....	28
2.4.3 JavaScript.....	28
2.5 MODELO VISTA CONTROLADOR.....	29
2.5.1 VENTAJAS Y DESVENTAJAS	29
2.5.2 MODELO	30
2.5.3 VISTA	30
2.5.4 CONTROLADOR	30
2.5.5 INTERACCIÓN DE LOS COMPONENTES	31
2.5.6 MVC en JAVA EE	31
2.6 RESUMEN	32
DISEÑO E IMPLEMENTACIÓN	33
3.1 CASOS DE USO	33
3.1.1 DIAGRAMAS DE CASOS DE USO	33
3.1.2 FUNCIONALIDADES OFRECIDAS	34
3.2 DINÁMICA DEL JUEGO.....	39
3.3 REQUISITOS	40
3.3.1 Requisitos de la aplicación	40

3.3.2 Requisitos de los profesores	40
3.3.3 Requisitos de los alumnos.....	43
3.4 MODELO DE DATOS.....	43
3.4.1 TABLAS.....	44
3.5 MODELOS	55
3.5.1 DIAGRAMA ENTIDAD-RELACIÓN	56
3.6 DIAGRAMAS DE FLUJO DEL FUNCIONAMIENTO	56
3.6.1 Inicio de la aplicación	56
3.6.2 Inicio sesión del usuario	57
3.6.3 Creación de grupos y registro de alumnos y/o profesores	57
3.6.4 Inicio sesión alumnos	60
3.6.5 Menú profesor	60
3.6.6 Creación competición.....	68
3.6.7 Realizar competición	69
3.6.8 Finalizar competición	70
3.7 RESUMEN DEL CAPÍTULO	71
VALIDACIÓN DE REQUISITOS.....	72
4.1 ACCESO A LA APLICACIÓN	72
4.1.1 Acceder a la aplicación (requisitos cumplidos: 1 y 3).....	72
4.1.2 Cambiar de contraseña (requisito cumplido: 2).....	73
4.2 MENÚ OPCIONES.....	73
4.3 GESTIONAR GRUPOS	74
4.3.1 Consultar alumnos de un grupo (requisitos cumplidos: 10, 29)	74
4.3.2 Editar grupos	75
4.3.3 Crear grupo.....	77
4.3.4 Borrar grupo (requisito cumplido: 16, 9, 33, 45)	77
4.4 GESTIONAR PROFESORES.....	78
4.4.1 Añadir profesor (requisitos cumplidos: 5, 42).....	78
4.4.2 Editar profesores	78
4.4.3 Ver profesores (requisitos cumplidos: 11, 43)	78
4.5 GESTIONAR COMPETICIONES.....	79
4.5.1 Listar todas las competiciones (requisito cumplido: 34).....	79
4.5.2 Listar competiciones abiertas (requisito cumplido: 35, 62, 63)	79
4.5.3 Listar competiciones cerradas (requisito cumplido: 36, 61)	79
4.5.4 Corregir preguntas finales alumnos (requisitos cumplidos: 41, 46, 31, 57, 58, 54, 55, 60)	79

4.5.5 Consultar preguntas corregidas por el profesor (requisitos cumplidos: 66, 67, 68) ..	81
4.5.6 Jugar competición (requisito cumplido: 41).....	81
4.5.7 Seguimiento de alumnos.....	85
4.5.8 Estadísticas competición	90
4.5.9 Crear competición (requisito cumplido: 32)	91
4.5.10 Borrar competición (requisito cumplido: 33).....	92
4.6 GUARDAR DATOS	92
4.6.1 Guardar preguntas (requisito cumplido: 52).....	92
4.6.2 Guardar revisiones (requisito cumplido: 59).....	93
4.6.3 Guardar correcciones	93
4.6.4 Guardar alumnos (requisito cumplido: 12)	93
4.6.5 Guardar profesores	93
4.6.6 Guardar competiciones (requisito cumplido: 40)	93
4.6.7 Guardar grupos (requisito cumplido: 13).....	93
4.6.8 Guardar respuestas juego (requisito cumplido: 53)	93
4.7 CASO DE ESTUDIO	93
4.8 RESUMEN	95
CONCLUSIONES Y TRABAJO FUTURO	96
5.1 CONCLUSIONES	96
5.2 TRABAJO FUTURO	96
PLANIFICACIÓN Y PRESUPUESTO	99
6.1 PLANIFICIACIÓN.....	99
6.2 PRESUPUESTO	101
APÉNDICE A	103
A.1 VALIDACIÓN DE REQUISITOS	103
A.1.1 INTRODUCCIÓN.....	103
A.1.2. ACCESO A LA APLICACIÓN.....	103
A.1.3 GESTIONAR GRUPOS.....	105
A.1.4 GESTIONAR PROFESORES.....	107
A.1.5 GESTIONAR COMPETICIONES	108
A.1.6 GUARDAR DATOS.....	117
BIBLIOGRAFÍA.....	118

ÍNDICE DE FIGURAS

1.1. Documento de seguridad.....	19
2.1. Proceso de revisión.....	21
2.2. Modelo Cliente-Servidor.....	26
2.3. Patrón MVC.....	30
2.4. Modelo MVC utilizado.....	32
3.1. Diagrama de Casos de Uso.....	35
3.2. Diseño de la base de datos.....	45
3.3. Tabla alumnos.....	45
3.4. Tabla alumnos competiciones.....	46
3.5. Tabla alumnos contador.....	47
3.6. Tabla competición.....	48
3.7. Tabla grupos.....	49
3.8. Tabla preguntas.....	49
3.9. Tabla preguntas cambiadas.....	50
3.10. Tabla preguntas jugadas.....	51
3.11. Tabla preguntas corregidas.....	52
3.12. Tabla profesores.....	52
3.13. Tabla profesores competición.....	53
3.14. Tabla profesores grupo.....	54
3.15. Tabla revisiones.....	54
3.16. Tabla respuestas.....	55
3.17. Relación de clases.....	57
3.18. Diagrama de flujo inicio aplicación.....	58
3.19. Inicio sesión administrador.....	58
3.20. Creación de un grupo.....	59
3.21. Registro de alumnos.....	59
3.22. Ejemplo de archivo CSV.....	59
3.23. Registro de profesores.....	60
3.24. Creación grupo a partir de archivo CSV.....	60

3.25. Inicio sesión alumno.....	61
3.26. Gestionar grupo.....	62
3.27. Gestionar competiciones.....	64
3.28. Cambiar contraseña profesor.....	65
3.29. Guardar datos.....	65
3.30. Gestionar grupos administrador.....	66
3.31. Gestionar profesores.....	66
3.32. Mis datos.....	67
3.33. Competiciones.....	67
3.34. Competiciones cerradas alumno.....	67
3.35. Resultados alumno.....	68
3.36. Preguntas corregidas profesor.....	69
3.37. Guía del juego.....	69
3.38. Creación competición.....	70
3.39. Realizar competición alumno.....	70
3.40. Realizar competición profesor.....	71
3.41. Finalización competición alumno.....	71
3.42. Finalización competición profesor.....	72
4.1. Acceso a la aplicación.....	73
4.2. Página principal del alumno.....	74
4.3. Página principal del profesor no administrador.....	75
4.4. Página principal del administrador.....	75
4.5. Seleccionar grupo.....	75
4.6. Listado alumnos.....	75
4.7. Importar archivo CSV.....	76
4.8. Listar alumnos del CSV y añadir al grupo.....	76
4.9. Añadir alumno manualmente.....	77
4.10. Editar alumno.....	77
4.11. Validar eliminar alumno.....	78
4.12. Crear grupo vacío.....	78
4.13. Borrar grupo.....	79

4.14. Fase corrección profesor.....	81
4.15. Lista de preguntas y correcciones.....	82
4.16. Formulario de las preguntas.....	83
4.17. Fase revisor.....	84
4.18. Fase corrector.....	85
4.19. Ir a fase juego.....	85
4.20. Fase juego.....	86
4.21. Lista alumnos que han hecho preguntas.....	87
4.22. Alumnos que han hecho revisiones.....	87
4.23. Alumnos que han hecho correcciones.....	88
4.24. Alumnos que han hecho fase juego.....	88
4.25. Fase realizada por cada alumno.....	89
4.26. Aciertos y fallos juego.....	89
4.27. Resultado alumnos.....	89
4.28. Resultado fase escritor.....	90
4.29. Lista preguntas fase juego.....	91
4.30. Lista preguntas exportar.....	92
4.31. Crear competición.....	92
4.32. Directorio donde se guardará el archivo.....	93
6.1. Diagrama de Gantt de la planificación.....	102
7.1. Acceso alumno.....	104
7.2. Acceso profesor.....	104
7.3. Contraseña incorrecta.....	105
7.4. Login no válido.....	105
7.5. Cambio de contraseña.....	105
7.6. Contraseña cambiada.....	105
7.7. Contraseña actual incorrecta.....	105
7.8. Contraseña no cumple condiciones.....	106
7.9. Opciones gestión grupo profesor administrador.....	106
7.10. Opciones gestión grupo profesor no administrador.....	106
7.11. No alumnos grupo.....	106

7.12. Opciones editar grupos.....	106
7.13. Lista alumnos a editar.....	107
7.14. Opciones a realizar.....	107
7.15. Opciones crear grupo.....	107
7.16. Grupo existente.....	107
7.17. Importar archivo CSV.....	107
7.18. Listar alumnos CSV y crear grupo.....	108
7.19. Opciones gestión profesor.....	108
7.20. Registro profesor.....	108
7.21. Editar profesor.....	108
7.22. Ver profesores grupo.....	109
7.23. No profesores grupo.....	109
7.24. Gestionar competiciones.....	109
7.25. Seleccionar grupo para listar competiciones.....	109
7.26. Lista de todas las competiciones del grupo 91.....	109
7.27. No hay competiciones cerradas.....	110
7.28. Acceso fase corrección profesor.....	110
7.29. Lista de alumnos.....	110
7.30. Lista preguntas alumno.....	110
7.31. No alumnos fase corrector.....	110
7.32. Pregunta de la fase escritor.....	110
7.33. Pregunta de la fase corrector.....	111
7.34. Lista de preguntas con corrección profesor.....	111
7.35. Puntuaciones guardadas.....	111
7.36. Alumno no preguntas validadas.....	111
7.37. Fin fase corrección profesor.....	111
7.38. Lista de competiciones a jugar.....	112
7.39. Mensaje acceso competición.....	112
7.40. Acceso a la fase escritor.....	112
7.41. Validación de preguntas.....	112
7.42. Acceso fase revisor.....	113

7.43. Acceso fase corrector.....	113
7.44. Corregir pregunta.....	113
7.45. Pregunta cambiada.....	113
7.46. Corrección validada.....	113
7.47. Acceso fase juego.....	114
7.48. Final de competición.....	114
7.49. Opciones de seguimiento.....	114
7.50. Lista preguntas iniciales y finales.....	114
7.51. Pregunta detallada.....	114
7.52. Opciones de revisiones.....	114
7.53. Lista preguntas revisiones obligatorias.....	115
7.54. Pregunta y revisión asociada.....	115
7.55. No alumnos revisiones.....	115
7.56. No alumnos revisiones opcionales.....	115
7.57. No alumnos correcciones.....	115
7.58. No alumno fase juego.....	116
7.59. Preguntas falladas.....	116
7.60. Resultados alumno.....	116
7.61. Resultado fase revisor.....	116
7.62. Puntuación fase corrector.....	116
7.63. No resultados fase corrector.....	116
7.64. No resultados juego inicial.....	117
7.65. No resultados juego final.....	117
7.66. Opciones de estadísticas.....	117
7.67. Alumno que ha acertado la pregunta.....	117
7.68. Descarga de las preguntas en CSV.....	117
7.69. Error fechas competición.....	118
7.70. Borrar competición deseada.....	118
7.71. Opciones para guardar datos.....	118
7.72. Datos guardados correctamente.....	118

ÍNDICE DE TABLAS

Tabla 2.1. Comparación bases de datos.....	25
Tabla 2.2. Comparación bases de datos.....	25
Tabla 6.1 Planificación dividida en tareas.....	100
Tabla 6.2 Horas empleadas.....	102
Tabla 6.3 Horas dedicadas agrupadas.....	103
Tabla 6.4 Material empleado.....	103
Tabla 6.5 Presupuesto final.....	103

Capítulo 1

INTRODUCCIÓN

En este capítulo de introducción se va a explicar la motivación del presente trabajo y los objetivos que se quieren conseguir con este, así como la estructura seguida en la memoria.

1.1 MOTIVACIÓN

Con la llegada del nuevo plan de estudios (Declaración de Bolonia) [1], las universidades españolas están siguiendo el modelo de evaluación continua [24], que consiste en realizar pruebas que se distribuyen a lo largo de todo el cuatrimestre, forzando a los alumnos a realizar distintos tipos de evaluaciones. A pesar de esto, la implicación y el aprendizaje no se ve aumentado en todos los casos. [2]

En muchos casos, los alumnos suelen abandonar la evaluación continua de las asignaturas que se basan más en realizar prácticas, ya que no llegan a entender los conceptos y no son capaces de aplicarlos. Esto presenta una dificultad añadida al aprendizaje a la que los alumnos se ven sometidos ya que es la primera vez que entran en contacto con este tipo de asignaturas.

Por todo esto, se ha podido observar que en muchas asignaturas, los alumnos simplemente intentan aprender lo necesario para aprobar e incluso a veces tratan de memorizar conceptos y algoritmos sin llegar a entenderlos del todo.

Diversos trabajos han llegado a la conclusión de que la realización de revisión entre pares o iguales es una de las metodologías que fomenta el interés de los alumnos profundizando en los conceptos, ya que pueden realizar críticas constructivas del trabajo de otros alumnos, ayudando de esta manera a aumentar el grado de motivación y la profundidad del aprendizaje por parte de los alumnos.

Además, se implementa en la parte final de la aplicación, un sistema de competición muy sencillo. A través de la competición, se pretende llegar a sacar lo mejor de una persona [3]: la perseverancia y el esfuerzo participan en este proceso y los alumnos se respetan a pesar de que alguno de ellos no haya experimentado la victoria.

En el presente trabajo fin de grado se implementa un sistema de revisión entre pares o iguales, con el objetivo no sólo de ayudar a los alumnos a interiorizar conceptos teóricos, sino que también de fomentar el desarrollo de su pensamiento crítico y capacidades de evaluación de cada alumno, capacidades necesarias en su futuro profesional.

Las preguntas generadas por los propios alumnos se usarán en una competición en la parte final de la aplicación. Con esto se persigue que los alumnos, en primer lugar, elaboren preguntas de más calidad y que además se interesen por el proceso de revisión,

elaborando más revisiones de las obligatorias, ya que cuantas más preguntas revisen más posibilidades tienen de que el sistema, en la parte de competición, les asigne preguntas que han revisado.

1.1.1 Entorno socio-económico

Como se ha dicho anteriormente, la universidad está siguiendo el modelo de evaluación continua. La realización de este tipo de evaluación supone el uso de recursos que son limitados. La escalabilidad y sostenibilidad se han vuelto críticas, y desafortunadamente la crisis del 2008 ha hecho que los recursos en las Universidades se hayan visto reducidos.

España posee un modelo de organización de la enseñanza centrada en los estudiantes más rígido que otros países. Por ejemplo, en muchas universidades de Estados Unidos los profesores usan a ayudantes o estudiantes de doctorado para tareas más rutinarias. Sin embargo, en España los docentes se suelen responsabilizar de todas las tareas relacionadas con los cursos, con grupos de laboratorio mucho más grandes que los de otras universidades extranjeras.

Racionalizar los recursos requeridos es fundamental para mejorar el aprendizaje. Por tanto, se necesitan métodos para satisfacer la necesidad de una evaluación formativa de calidad sostenible y, además, para implementar de forma adecuada y continua la evaluación sumativa.

Entendiendo como evaluación sumativa [24] aquella que se desarrolla al final del proceso enseñanza-aprendizaje mientras que la evaluación formativa está centrada en el proceso de enseñanza para mejorarlo durante su fase de desarrollo.

Por todo ello, surge la revisión entre pares o iguales como método para fomentar el aprendizaje de las habilidades que requiere una ingeniería.

1.2 OBJETIVOS

El objetivo principal de este trabajo fin de grado es la creación de un juego educativo basado en la revisión por pares. Lo que se pretende con este juego es que los alumnos muestren interés por aprender la asignatura y que desarrollen una estrategia de aprendizaje.

Este juego se usará como herramienta educativa, del cual se aprovechará el carácter competitivo que presenta para incentivar su uso al alumno y que de esta forma aprenda el contenido teórico de la asignatura a medida que avanza en el juego, mediante la implementación de revisiones.

Este objetivo se materializa en los siguientes sub-objetivos concretos:

1. Estudio de herramientas web para la realización de la aplicación.
2. Estudio de las posibles tecnologías que se utilizarán para desarrollar el juego.
3. Estudio de sistemas de revisión entre pares para tener conocimiento de las ventajas que supone dicha metodología.

4. Implementación del sistema creado.
5. Realización de pruebas para comprobar el correcto funcionamiento de la aplicación.
6. Caso de estudio en una asignatura concreta: en el momento de la escritura de esta memoria, esta aplicación está siendo usada en una asignatura concreta con el fin de validar si su uso fomenta realmente el interés y motivación del alumnado.

Además, posteriormente a la finalización de este trabajo fin de grado, se analizarán los datos obtenidos con el fin de valorar si realmente el uso de la revisión entre pares ha conseguido los objetivos esperados.

1.3 MARCO REGULADOR

Como se ha ido describiendo anteriormente, este trabajo fin de grado va a manejar datos pertenecientes a un número de alumnos. Por ello, en este apartado se hablará de acerca del marco regulador relacionado con el manejo de datos que está presente en la aplicación que se va a crear. Esto conlleva a la mención de la LOPD (Ley Orgánica de Protección de Datos) [4].

La LOPD es una Ley Orgánica española cuyo objetivo es establecer las medidas y necesarias para poder garantizar la seguridad que deben tener los ficheros, los centros de tratamiento, locales, programas, sistemas y los encargados de los tratamientos de los datos de carácter personal.

El responsable de un fichero o tratamiento es aquella entidad, persona y órgano administrativo que define la finalidad, el contenido y el uso del tratamiento de los datos personales [5].

Hay que mencionar la Agencia Española de Protección de Datos que se creó a partir de la aparición de dicha ley. Se trata de una organización que se encarga del cumplimiento de esta normativa a nivel nacional.

Los ficheros hacen referencia al conjunto de datos personales independientemente de cual se la forma en la que se ha creado, su organización y acceso. Si alguien crea estos ficheros, deberá inscribirlos en el Registro General de Protección de Datos (RGPD) [6].

Además, esta ley presenta tres niveles de seguridad que son BÁSICO, MEDIO Y ALTO. Se clasifican de esta forma ya que dependen de la naturaleza de la información que se trate, en relación con la mayor o menos necesidad que tenga para garantizar la confidencialidad y la integridad de la información.

A continuación se presentan los ficheros y tratamientos a los que corresponde aplicar las medidas de seguridad relacionadas a cada uno de los niveles descritos anteriormente:

- Básico: se trata de ficheros que contengan datos personales tales como nombres, apellidos, códigos postales, domicilios, ciudad o teléfono.

- Medio: se trata de ficheros o tratamiento con datos relativos a la comisión de infracciones administrativas o penales, datos de Administraciones tributarias, de Entidades Gestores y Servicios Comunes de Seguridad Social y de entidades financieras.
- Alto: se trata de ficheros o tratamiento con datos de ideología, información de tipo política, religión, creencias, origen racial, vida sexual o salud.

Una vez vista la clasificación anterior, podemos determinar que el nivel que se va a usar en el presente trabajo fin de grado es el básico, ya que utilizaremos los nombres y apellidos de alumnos.

El siguiente paso que se debe hacer, es inscribir el fichero en el RGPD. Se registrará asignando un nombre al fichero y un identificador que es el número de alta cuando se inscribe el fichero y donde estará relacionada la información que se desea proteger.

Una vez que esté realizado el registro, se lleva a cabo el documento de seguridad que detalla el nivel de datos a proteger y el tipo que sea va a inscribir en el fichero.

En la figura 1.1 se muestra el documento para observar como delimitar de una forma precisa el contenido que se desea registrar.

NOMBRE DEL FICHERO		ORIGEN	
Usuario de la Web		El Propio Interesado	<input checked="" type="checkbox"/>
		Registros Públicos	<input type="checkbox"/>
		Otras Personas Físicas	<input type="checkbox"/>
		Entidad Privada	<input type="checkbox"/>
		Fuentes Accesibles al Público	<input type="checkbox"/>
		Administraciones Públicas	<input type="checkbox"/>
Sistema de Tratamiento Automatizado <input type="checkbox"/> Manual <input type="checkbox"/> Mixto <input checked="" type="checkbox"/>		Nivel de Seguridad del Fichero Básico <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Alto <input type="checkbox"/>	
Finalidades Previstas		Colectivos Interesados	
Gestión de Clientes Contable, Fiscal y Administrativa	<input type="checkbox"/>	Empleados	<input checked="" type="checkbox"/>
Recursos Humanos	<input type="checkbox"/>	Clientes y Usuarios	<input checked="" type="checkbox"/>
Gestión de Nóminas	<input type="checkbox"/>	Proveedores	<input type="checkbox"/>
Prevención de Riesgos Laborales	<input type="checkbox"/>	Asociados/Miembros	<input type="checkbox"/>
Prestación de Servicios de Solvencia Patrimonial y Crédito	<input type="checkbox"/>	Propietarios/Arrendatarios	<input type="checkbox"/>
Cumplimiento/Incumplimiento de Obligaciones Dinerarias	<input type="checkbox"/>	Pacientes	<input type="checkbox"/>
Servicios Económicos Financieros y Seguros	<input type="checkbox"/>	Estudiantes	<input type="checkbox"/>
Análisis de Perfiles	<input type="checkbox"/>	Personas de Contacto	<input type="checkbox"/>
Publicidad y Prospección Comercial	<input type="checkbox"/>	Padres/Tutores	<input type="checkbox"/>
Prestación de Servicios de Comunicación Electrónica	<input type="checkbox"/>	Representante Legal	<input type="checkbox"/>
Guías/Repertorios de Servicios de Comunicaciones Electrónicas	<input type="checkbox"/>	Solicitantes	<input type="checkbox"/>
Comercio Electrónico	<input type="checkbox"/>	Beneficiarios	<input type="checkbox"/>
Prestación de Servicios de Certificación Electrónica	<input type="checkbox"/>	Cargos Públicos	<input type="checkbox"/>
Gestión de Asociados/Partidos Políticos/Iglesias/Sindicatos (sin lucro)	<input type="checkbox"/>	Otros Colectivos	<input type="checkbox"/>
Actividades Asociativas, Culturales, Recreativas, Deportivas y Sociales	<input type="checkbox"/>		
Gestión de Asistencia Social	<input type="checkbox"/>		
Educación	<input type="checkbox"/>		
Investigación Epidemiológica y Actividades Analógicas	<input type="checkbox"/>		
Gestión y Control Sanitario	<input type="checkbox"/>		
Historial Clínico	<input type="checkbox"/>		
Seguridad Privada	<input type="checkbox"/>		
Seguridad y Control de Acceso a Edificios	<input type="checkbox"/>		
Video Vigilancia	<input type="checkbox"/>		
Fines Estadísticos, Históricos o Científicos	<input type="checkbox"/>		
Otros tipos de finalidad	<input type="checkbox"/>		

Figura 1.1 Documento de seguridad

1.4 ESTRUCTURA DE LA MEMORIA

Esta sección recoge toda la información relacionada con la realización del trabajo fin de grado que se expone que se divide en diferentes capítulos. Cada uno de estos capítulos posee una determinada información que se relaciona entre sí y sirve para describir un determinado aspecto del presente trabajo.

A continuación se va a describir el tipo de información que contiene cada capítulo, que servirá para una mejor comprensión futura de cada uno de ellos.

- **Introducción:** en este capítulo, como se ha visto, se han presentado las ideas del trabajo de manera básica y se ha procedido a explicar los objetivos y la motivación de la realización de este.
- **Estado del arte:** en este capítulo se analizarán los aspectos que han influido para elegir el entorno de trabajo usado en la realización del proyecto. Se van a tratar las diferentes tecnologías y lenguajes de programación que se han empleado explicando de forma breve las características y los usos de cada tecnología.
- **Diseño e implementación:** en este capítulo se explicará la composición principal del trabajo. Se profundizará en aspectos técnicos y se detallarán los requisitos que tiene que cubrir la aplicación para cumplir los objetivos descritos anteriormente.
- **Validación de los requisitos:** en este capítulo se llevará a cabo una demostración para ver que la aplicación cumple los requisitos planteados en el capítulo anterior y que son necesarios para cumplir los objetivos. Para la demostración se emplearán capturas de imágenes pertenecientes a la aplicación y una explicación de cada caso. Además, se hablará sobre el la recepción del caso de estudio.
- **Conclusiones y trabajo futuro:** en este capítulo se destacan las conclusiones sobre todo el proceso de desarrollo de la aplicación que se realiza en el presente trabajo fin de grado. Además, se presentan posibles desarrollos futuros que podrían ser mejoras o ampliación de la misma aplicación.
- **Planificación y presupuesto:** en la planificación se tratarán las tareas del trabajo realizado en el proyecto y se hace una estimación temporal de cada una de ellas. El presupuesto describe las horas dedicadas a cada tarea, así como el coste del material

Capítulo 2

ESTADO DEL ARTE

En este capítulo se va a hablar sobre en qué consiste la revisión entre pares que es utilizada para el desarrollo de la aplicación.

Además, se hará un estudio comparativo de las tecnologías que se pueden utilizar para el desarrollo del trabajo y en algún caso se dará una explicación del por qué se ha utilizado una tecnología frente a otras.

En resumen, este capítulo sirve para poder comprender mejor la realización de los siguientes capítulos y explicar la temática que sigue la aplicación.

2.1 REVISIÓN ENTRE PARES

La revisión entre pares consiste en la evaluación del trabajo de un compañero con nivel similar de conocimiento, por lo que también se le puede denominar revisión entre iguales. Es un proceso de evaluación donde los estudiantes juzgan la calidad, nivel, valor o resultados de aprendizaje de otros compañeros que tengan un nivel similar [7].

En la figura 2.1 se puede apreciar el ciclo que sigue este tipo de revisiones.

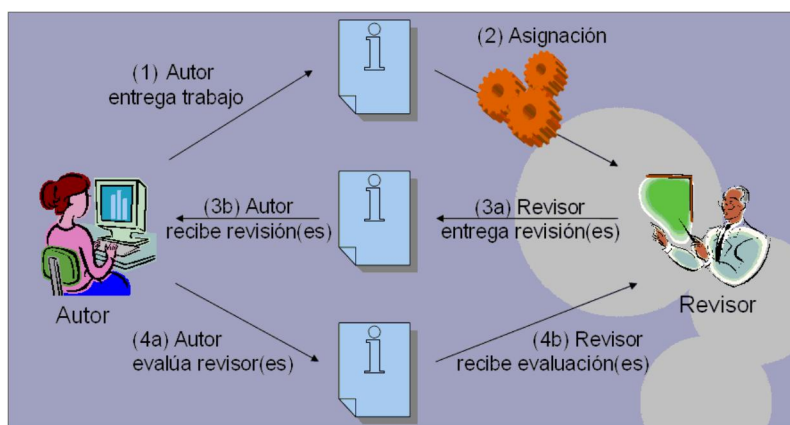


Figura 2.1 Proceso de revisión.

1. Desarrollo del trabajo. Es la primera fase y es donde los alumnos realizan el trabajo, ya sea individualmente o por grupos.
2. Asignación de los revisores. El trabajo es asignado a un conjunto de revisores que estará determinado por el profesor.
3. Revisión de los trabajos. Cada revisor debe evaluar los trabajos que tiene asignados de la fase anterior. Esta evaluación suele ser anónima. Una vez realizada la revisión, es devuelta al autor del trabajo.

4. Evaluación de las revisiones. El autor valora la realimentación recibida. En esta evaluación debe valorar su conformidad con los comentarios recibidos, si la revisión ha sido hecha de forma exhaustiva y la utilidad de dicha revisión.

Este ciclo puede repetirse las veces que sean necesarias. Como se ha podido ver, está formado por cuatro fases, de las cuales la fase de asignación de los revisores es oculta y transparente para los alumnos.

Los beneficios que se pretenden conseguir haciendo uso de esta metodología se pueden dividir en:

- A. Objetivos específicos de la asignatura
 - Mayor motivación por parte del alumno.
 - Promover el análisis de la asignatura.
 - Impulsar a que los alumnos hagan un trabajo continuo para conseguir un aprendizaje de calidad
 - Revisión de soluciones alternativas.
 - Proporcionar realimentación a los alumnos para resolver dudas.
 - Afianzar los conocimientos del alumnado y profundizar en la materia a través de la aplicación práctica.
- B. Objetivos transversales
 - Promover la responsabilidad de un rol más autónomo.
 - Potenciar el desarrollo de habilidades que no son técnicas pero son imprescindibles por parte del alumno (trabajo en equipo, capacidad de síntesis, razonamiento y justificación...).

2.2 SISTEMA DE COMPETICIÓN

El trabajo fin de grado que se presenta, es la realización de una aplicación que se basará en un sistema de competición entre alumnos de una asignatura.

Se ha basado en un sistema de competición, ya que puede llegar a sacar lo mejor de una persona. La perseverancia y el esfuerzo participan en este proceso y los alumnos se respetan a pesar de que alguno de ellos no haya experimentado la victoria [3].

La idea principal es motivar a los alumnos a aprender conocimientos de la asignatura y fomentar el trabajo diario por parte del alumno, de tal manera que consiga un aprendizaje progresivo. También se intenta crear una motivación extra por parte del alumno para que se esfuerce al máximo en realizar bien las preguntas, ya que estas se encontrarán en la evaluación.

Realizar las pruebas de una forma más interactiva y divertida ayudará a que el alumno aprenda y que tenga un buen concepto de la asignatura en cuestión.

2.3 TECNOLOGÍAS DEL SERVIDOR

En el presente trabajo se utilizará como servidor Apache Tomcat y el Sistema Gestor de Base de Datos será MySQL. La lógica que permite la gestión de los datos y la conexión

de las peticiones de los clientes con el servidor estará implementada en JAVA. Además, la aplicación seguirá la estructura modelo vista controlador (MVC).

A continuación se presentan las tecnologías empleadas y la comparación de estas con otras.

2.3.1 Sistema Gestor de Base de Datos

El propósito de dicho sistema es el manejo claro, sencillo y ordenado de un conjunto de datos que se convertirán en información relevante para el usuario.

Los datos que se guardan pertenecen a un mismo contexto y serán utilizados posteriormente por el usuario.

Este trabajo se va a centrar en el uso de un modelo de datos relacional (SGBDR) ya que es el modelo que se suele emplear en la actualidad para administrar datos dinámicamente. En este tipo de modelo, la base de datos es representada como una colección de tablas donde todos los datos guardan relación entre ellos.

Para este tipo de bases de datos se suele utilizar SQL (Structured Query Language) como lenguaje para consultas de alto nivel, frente a las bases de datos No-SQL o base de datos no relacionales.

Este tipo de bases de datos son diferentes del modelo clásico del SGBDR principalmente en que no usan SQL como lenguaje de consultas y en que los datos almacenados no requieren estructuras fijas como tablas.

A continuación, se van a presentar algunos de los tipos de bases de datos que han sido considerados antes de la realización de la aplicación.

2.3.1.1 PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional de código abierto [9]. Es objeto-relacional ya que incluye características propias de la orientación a objetos, como lo son la herencia, funciones, tipos de datos, disparadores e integridad transaccional.

Se inició en 1982 en la universidad de Berkeley y en 1985 se crea el proyecto POSTGRES o POSTGRES basado en las ideas de este. Posteriormente, empezó a utilizar SQL recibiendo de esta manera su nombre actual.

Algunas de las características que presenta son:

- Utiliza modelo cliente-servidor y usa multiprocesos en vez de multihilos para poder garantizar estabilidad.
- Permite la declaración de funciones propias, así como de disparadores.
- Soporta distintos tipos de datos y permite la creación de tipos propios.
- Incluye herencia entre tablas, por lo que se trata de un gestor objeto-relacional.
- Permite gestionar diferentes usuarios.

- Está disponible para la mayoría de sistemas operativos y dispone de API's en varios lenguajes de programación (C / C ++, Java, PHP)
- Alta concurrencia, controlado por MVCC (Acceso concurrente multiversión)
- Su administración se basa en usuarios y privilegios.
- Integridad transaccional.

2.3.1.2 Oracle

Se trata de un sistema gestor de base de datos objeto-relacional y utiliza un lenguaje procedural (PL/SQL) donde los datos son tratados como unidad [10]. Una de sus principales funciones es el uso de recursos del sistema informático para poder aprovechar su rendimiento al máximo en situaciones cargadas con mucha información, combinándose de esta forma rendimiento y estabilidad.

Oracle presenta las siguientes características:

- Escalabilidad, ya que soporta gran cantidad de datos.
- Alto rendimiento en las transacciones.
- Coste muy alto.
- Estabilidad.
- Su instalación suele ser complicada
- Se puede acceder sin necesidad de abrir una base de datos
- Autogestión de la integridad de los datos
- Multiplataforma
- Posee un registro de cambios
- Replicación
- Gestiona la seguridad

2.3.1.3 MySQL

MySQL es un sistema de gestión de base de datos relacional (SGBDR) [11], multihilo y multiusuario, rápida, fácil y segura de utilizar con gran cantidad de usuarios que la utilizan.

Es desarrollada por MySQL AB, que tiene los derechos de código y desarrolla software libre bajo licencia GNU GPL [3] para cualquier uso compatible con esta licencia. En caso de que haya empresas que quieran utilizarlo en productos privativos, deberán comprar una licencia específica para este uso. Está desarrollado en su mayor parte en ANSI C.

Es una base de datos rápida en lectura aunque puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. Por otro lado, en las aplicación web no suele haber demasiada concurrencia en los datos y si una gran cantidad de lectura en el entorno, lo que hace que MySQL sea ideal para este tipo de aplicaciones.

Dicha base de datos presenta las siguientes características:

- Gracias a su implementación multihilo, aprovecha la potencia de sistemas multiprocesador generando mayor velocidad.
- Es capaz de soportar una gran cantidad de tipos de datos en las columnas.
- Disponibilidad de gran cantidad de plataformas y sistemas.
- Gran cantidad de librerías para permitir su uso en gran cantidad de lenguajes de programación ya que dispone de API's para distintos lenguajes (C, C++, Java)
- Conectividad segura ya que realiza gestión de usuarios y contraseñas.
- Su instalación es fácil y su configuración sencilla.
- Su utilización es gratuita.
- Robusto soporte transaccional que garantiza la integridad de los datos en todo momento.
- Escalable y flexible ya que se adapta a las necesidades de tamaño.
- Replicación.
- Alto rendimiento.
- Transacción y claves foráneas.

2.3.1.4 Comparación BBDD

A continuación se presentan en las tablas 2.1 y 2.2 una comparativa de las características más importantes de las bases de datos anteriormente citados. La tabla presentará un “+” si cumple con la característica descrita y con “++” si dicha característica destaca sobre las demás.

	Alto rendimiento	Velocidad	Robusto	Control de acceso	Limitación del tamaño registros	Bajo consumo recursos
PostgreSQL		+	+	+	+	
Oracle	++	+	++	+		
MySQL	+	++	++			+

Tabla 2.1. Comparación bases de datos

	Seguridad	Escalable	Transacciones	Multiplataforma	Definición de funciones
PostgreSQL	+	+	+	+	+
Oracle	++	++	+	++	++
MySQL		+	+	++	

Tabla 2.2. Comparación bases de datos

Como se puede apreciar en las tablas, Oracle presenta mejores características que los demás pero al tener un coste muy elevado, tiene que ser descartado. Por lo tanto, se podrá elegir entre PostgreSQL y MySQL. En este caso se ha elegido utilizar MySQL para el desarrollo de la aplicación, ya que presenta mejores características en cuanto a la velocidad, el alto rendimiento y la robustez. Además, como se ha comentado anteriormente, es la más usada en las aplicaciones web debido a su gran cantidad de lectura.

2.3.2 Servidor Web

La mayoría de las aplicaciones web se basan en un modelo Cliente-Servidor. Se trata de un modelo de aplicación distribuida donde las tareas se reparten entre los proveedores de recursos o servicios, que llamaremos servidores y los que demandan dichos recursos, que llamaremos clientes [12]. El cliente realizará una serie de peticiones y el servidor dará una respuesta a estas, aplicando una lógica programada determinada. La figura 2.2 presenta un esquema del modelo donde el cliente envía una petición y el servidor manda la respuesta.

Un servidor web es un programa informático que procesa una aplicación del lado del servidor y realiza conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente, generando una respuesta. Cuando detecta una conexión, el servidor espera a recibir una petición en el formato adecuado. La petición realizada y la respuesta que se da se encapsulan siguiendo el protocolo HTTP.

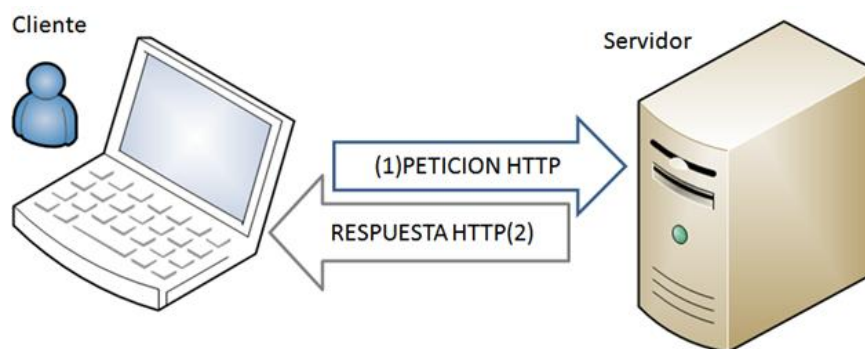


Figura 2.2 Modelo Cliente-Servidor

Los usuarios ejecutarán la parte cliente de la aplicación. Por lo general, un servidor puede tratar múltiples peticiones a la vez (varios clientes a la vez).

El servidor que se ha utilizado para realizar esta aplicación ha sido Apache Tomcat.

2.3.2.1 Apache Tomcat

Apache tomcat (o simplemente tomcat) sirve como un contenedor de servlets. Tomcat implementa las especificaciones de los servlets y de los JavaServer Pages (JSP) [13].

Es un contenedor web con soporte de servlets y JSPs. No es un servidor de aplicaciones e incluye el compilador Jasper, que compila los JSPs y los convierte en servlets.

Tomcat puede funcionar como un servidor web independiente y es usado como servidor web autónomo en entornos que tengas un alto nivel de tráfico y alta disponibilidad.

Al haber sido escrito en Java, función con cualquier sistema operativo que disponga de una máquina virtual de Java.

2.3.2.1.1 Servidor apache

Se trata de un servidor HTTP de código abierto que implementa el protocolo HTTP/1.1 y la noción de sitio virtual.

El servidor apache es desarrollado por usuarios bajo la supervisión de la Apache Software Foundation dentro del proyecto HTTP Server [14]. Es el servidor HTTP más usado en internet. Dentro de sus ventajas está el que es modular, de código abierto, multiplataforma, extensible y popular.

Este servidor presenta bases de datos de autenticación y negociado de contenido.

Apache reside en un equipo conectado a la red, que acepta las peticiones que recibe de los navegadores Web y otros clientes en equipos remotos, y responde enviando el contenido solicitado. Es usado para tareas donde el contenido necesita ser puesto a disposición de una forma segura y confiable.

2.3.3 Lenguaje de programación

A continuación se van a presentar varias opciones de lenguaje de programación, aunque el lenguaje empleado para toda la lógica de procesamiento será Java.

2.3.3.1 Java

La sintaxis de java deriva de C y C++ aunque tiene menos utilidades de bajo nivel que cualquiera de ellos [15]. Las aplicaciones de Java son compiladas a bytecode (clase Java) que puede ejecutarse en cualquier máquina virtual java, independientemente de la arquitectura de la computadora subyacente.

Es un lenguaje de programación concurrente y orientado a objetos y basado en clases que se diseñó para tener tan pocas dependencias de implementación como fuera posible.

Su principal objetivo es permitir que los desarrolladores escriban el programa una vez y puedan ejecutarlo en cualquier dispositivo, lo que indica que el código que es ejecutado en una plataforma no tiene que ser recompilado para ejecutarse en otra.

Además, hay que destacar el hecho de que posee un recolector de basura que se trata de un mecanismo implícito que reserva, libera y evita fugas de memoria. El programador el momento de la creación de objetos y el entorno en tiempo de ejecución es el que gestiona el ciclo de vida de los objetos. El programa puede tener localizado un objeto mediante una referencia. Cuando ya no quedan referencias, el recolector de basura borra el objeto, liberando de esta manera la memoria que ocupaba y previniendo posibles fugas. A pesar de esto, es posible que haya fugas de memoria si el código almacena

referencias a objetos que ya no son necesarios. Resumiendo, el recolector de basura permite una creación y eliminación de objetos fácil y mayor seguridad.

2.3.3.2 Python

Python es un lenguaje de programación interpretado cuya filosofía se centra en una sintaxis que favorezca un código legible [16]. El intérprete de Python compila a bytecode el programa, lenguaje máquina optimizado para una máquina virtual de Python.

Se trata de un lenguaje de programación multiparadigma, lo que significa que los programadores pueden utilizar varios estilos de programación: orientada a objetos, programación imperativa y, en menor medida, programación funcional. Otros paradigmas son soportados mediante el uso de extensiones.

Otra característica importante de Python es que usa tipado dinámico y es multiplataforma, además de la resolución dinámica de nombres.

Para la administración de la memoria se usa el conteo de referencias. Un objetivo del diseño del lenguaje es la facilidad de extensión. Se pueden escribir nuevos módulos en C o C++ y puede incluirse en aplicaciones que necesiten una interfaz programable.

2.3.2.3 PHP

Es un lenguaje de programación de uso general de código del lado del servidor diseñado para el desarrollo web de contenido dinámico [17].

Fue uno de los primeros lenguajes que se podían incorporar directamente en el documento HTML y así evitar llamar a un archivo externo que procesara los datos. Su código es interpretado por un servidor web con un módulo de procesador PHP que genera la página web. Incluye una interfaz de línea de comando para ser utilizada en aplicaciones gráficas independientes.

Es considerado uno de los lenguajes más flexibles, potente y de alto rendimiento, por lo que ha atraído el interés de múltiples sitios con gran demanda de tráfico, como por ejemplo Facebook, para utilizarlo como tecnología de servidor.

2.4 TECNOLOGÍAS DEL CLIENTE

En este apartado se estudiarán las tecnologías correspondientes con la parte del Cliente, que es totalmente independiente al Servidor.

El cliente está formado por una serie de estructuras programáticas que le permiten ejecutar documentos e instrucciones que han sido cargados en él. Estas tecnologías permiten que los procesos se ejecuten del lado del cliente para que de esta forma, se aceleren las conexiones y no haya sobrecarga en los servidores.

A continuación se van a describir las tecnologías empleadas en el trabajo y otras posibles.

2.4.1 HTML

HTML (HyperText Markup Language) es un lenguaje de marcado utilizado para la elaboración de páginas web [18]. Define una estructura básica y un código (HTML) para la definición de contenido de una página web, como por ejemplo texto, imágenes, videos, etc.

Se considera el lenguaje web más importante siendo su invención fundamental en la aparición, desarrollo y expansión de la World Wide Web. Todos los navegadores actuales han adoptado este estándar ya que se ha impuesto en la visualización de páginas web.

El lenguaje HTML se basa en la diferenciación. Para añadir un elemento que es externo a la página, se hace una referencia a la ubicación de dicho elemento mediante texto, por lo que la página web sólo contiene texto y recae en el navegador web el deber de unir todos los elementos y cargarlos para poder visualizar la página final.

2.4.2 CSS

CSS (Cascading Style Sheets) es un lenguaje que sirve para describir las propiedades de diseño de un contenido en particular de una estructura, que por lo general se trata de HTML [19] [20].

Al utilizar CSS lo que se pretende es separar el contenido de la página y la información sobre su aspecto, permitiendo a los desarrolladores tener el control sobre el estilo y el formato de sus documentos.

Dentro de la página HTML se crea una zona en la que se incluye toda la información relacionada con los estilos de esta, lo que permite al programador generar un código limpio y de fácil acceso. La información puede estar definido en un archivo externo o dentro de la propia página, donde estará limitada mediante la etiqueta <STYLE>.

Con CSS se están escribiendo una serie de reglas cada una de las cuales se asocia a un elemento o parte de la página que deberá respetar estas reglas de diseño que se han establecido.

2.4.3 JavaScript

Es un lenguaje de programación interpretado, orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico [21].

Con JavaScript se pretende que el usuario pueda interactuar dinámicamente con la página web en cuestión, de esta manera JavaScript y HTML pueden interactuar, lo que permite a los programadores utilizar contenido dinámico [4].

La programación se realiza dentro del mismo archivo HTML, es decir, las funciones se escribirán dentro del mismo archivo donde se encuentre el código HTML.

Se utiliza normalmente en el lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

JavaScript se diseñó basándose en una sintaxis similar a C, y aunque adopta nombres y convenciones del lenguaje de programación Java, no están relacionados, teniendo propósitos y semánticas diferentes.

Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

2.5 MODELO VISTA CONTROLADOR

El presente trabajo va a seguir un patrón MVC. Modelo vista controlador (MVC) se trata de un patrón de arquitectura de software que separa los datos y la lógica de negocio de la interfaz de usuario y el módulo que se encarga de gestionar los eventos y las comunicaciones [22]. Su objetivo es desacoplar la vista y el modelo, por lo que la vista no puede acceder directamente al modelo.

Lo que hace este patrón es dividir la aplicación en tres componentes distintos que son, como se ha mencionado anteriormente, modelo, vista y controlador. Por un lado define componentes para representar la información y por otro lado para interactuar con el usuario. Está basado en la reutilización de código y la separación de conceptos.

La figura 2.3 representa un esquema de dicho patrón.

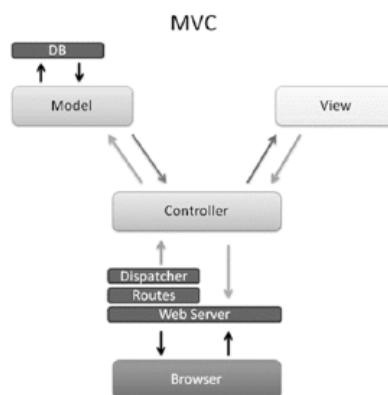


Figura 2.3 Patrón MVC

2.5.1 VENTAJAS Y DESVENTAJAS

Se presentan las ventajas y desventajas que tiene el utilizar este patrón [23]:

➤ VENTAJAS

- La implementación se realiza de forma modular
- Las vistas muestran siempre información actualizada, ya que el modelo lo realiza automáticamente.

- Si hay alguna modificación que afecte al dominio, implica solo un cambio en el modelo y en las vistas de este y no en todo el mecanismo de comunicación y de actualización entre modelos.
- Las modificaciones que sufren las vistas no afectan al modelo del dominio ni a su tratamiento, sólo a la representación de la información.
- Las aplicaciones que implementan dicho patrón, presentan una gran extensibilidad y mantenimiento.

➤ DESVENTAJAS

- Se requiere mayor tiempo inicial en el desarrollo de la aplicación que presente dicho patrón. Requiere que el programador desarrolle un mayor número de clases.
- Requiere la existencia de una arquitectura inicial sobre la que se construyen clases e interfaces para modificar y comunicar los módulos de la aplicación. La arquitectura debe incluir un mecanismo de eventos para proporcionar las notificaciones del modelo, una clase modelo, una clase Vista y una clase Controlador genéricas para la realización para las tareas de comunicación.
- Al ser orientado a objetos, su implementación es costosa y difícil para lenguajes que no cumplen dicho paradigma.

2.5.2 MODELO

Representa la información con la cual el sistema opera. Gestiona todos los accesos a dicha información, tanto las actualizaciones como las consultas, implementando los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación.

Es responsable de la recuperación de datos, que los convierte en conceptos significativos para la aplicación, así como de su procesamiento, validación, asociación y cualquier tarea que tenga que ver con la manipulación de datos.

Envía a la vista la información que se le solicita para ser mostrada. Las peticiones llegan a él a través del controlador.

2.5.3 VISTA

Hace una representación del modelo estando separada de los objetos de este. Es responsable del uso de cualquier información de la cual dispone para producir una interfaz de presentación de cualquier petición.

Presenta el modelo adecuado para interactuar con la interfaz de usuario, requiriendo del modelo la información que debe representar como salida.

2.5.4 CONTROLADOR

Gestiona las peticiones del usuario. Es el responsable de responder a la información que se solicita con la ayuda del modelo y de la vista.

Invoca peticiones al modelo cuando se hace alguna solicitud sobre la información. Además, puede enviar comandos a la vista que tenga asociada si en algún momento se solicita un cambio en la forma en la que se presente el modelo.

Por todo esto, se puede concluir que el controlador es un intermediario entre el modelo y la vista.

2.5.5 INTERACCIÓN DE LOS COMPONENTES

El flujo que se sigue normalmente con este patrón es:

1. El usuario interactúa con la interfaz de usuario realizando una solicitud.
2. El controlador recibe la notificación de la acción solicitada por el usuario.
3. El controlador accede al modelo modificándolo adecuadamente a la acción que ha solicitado el usuario.
4. El controlador manda a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene los datos del modelo y genera la interfaz apropiada donde se representan los cambios en el modelo. El modelo no puede tener conocimiento directo de la vista.
5. Una vez terminado esto, la interfaz espera nuevas acciones por parte del usuario para empezar otra vez el ciclo.

2.5.6 MVC en JAVA EE

El modelo que se utilizará para esta aplicación contendrá un servlet controlador. Esto quiere decir que todas las solicitudes que se hagan se enviarán a un servlet que actuará como controlador. Este servlet lee los parámetros de entrada, los valida y decide el siguiente paso a realizar dependiendo de los parámetros que se han recibido, la URL a la que el usuario quiere acceder y el estado anterior de la aplicación. La figura 2.4 hace referencia al proceso descrito.

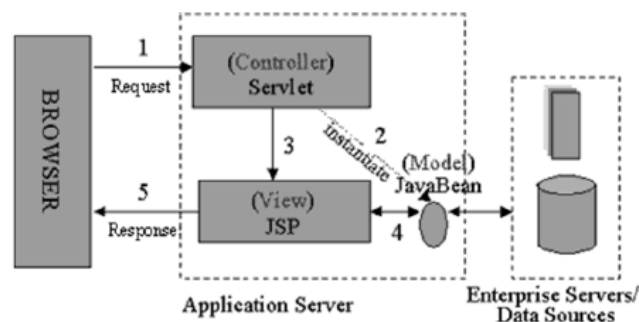


Figura 2.4 Modelo MVC utilizado

El controlador ejecuta la lógica de negocio que haya decidido. Si es sencilla la puede hacer el mismo aunque normalmente suele usarse otra clase Java, ejecutando una función de esta clase a la que se le pasan los parámetros que se han recibido en la solicitud. Una vez ejecutada la función, el controlador recibe el resultado de dicha ejecución.

Cuando se ha obtenido el resultado, el controlador decide a que vista (JSP) tiene que redirigir la petición, hará un forward al JSP y este se ejecutará, accediendo a la información y la formateará adecuadamente para poder presentársela al usuario.

2.6 RESUMEN

Se puede concluir que el uso de la revisión entre pares ayudará a potenciar el interés y el aprendizaje del alumno, así como habilidades personales como lo son la capacidad de trabajar en equipo y el razonamiento.

De igual manera, el uso de un sistema de competición ayudará a aumentar motivación del alumno para que se esfuerce al máximo.

Por otra parte, en el desarrollo de aplicaciones web existe gran variedad de tecnologías que se pueden implementar, por lo que es necesario un estudio a priori de los requisitos de la aplicación para poder elegir aquellas herramientas que ayuden a realización de estos. En este apartado se ha estudiado algunas de estas tecnologías para su posterior elección de uso.

Capítulo 3

DISEÑO E IMPLEMENTACIÓN

En este capítulo se va a detallar de una forma más técnica en qué consiste la aplicación y su implementación, explicando los pasos más importantes para su desarrollo. Se centrará principalmente en explicar la estructura de los datos y la lógica que se ha seguido para su ejecución. Además se hará un pequeño estudio de los casos de uso y se explicarán los requisitos que debe seguir la aplicación.

3.1 CASOS DE USO

Los casos de uso nos proporcionan los diferentes escenarios donde definimos la relación de cada uno de los usuarios con el sistema.

En este sistema van a interactuar tres usuarios diferentes que son los actores de este proyecto: alumno, profesor y administrador.

El alumno será capaz de jugar a la competición realizando las cuatro fases de las que se compone, creando preguntas, puntuando y comentándolas, corrigiendo dichas preguntas y jugándolas. También será capaz de ver los resultados obtenidos en cada competición que haya jugado así como las preguntas que haya corregido el profesor, cambiar de contraseña si lo desea y acceder a sus propios datos. El profesor será capaz de gestionar los grupos, gestionar alumnos y gestionar las competiciones, así como de cambiar de contraseña si lo desea. Y por último el administrador, que también podrá ser profesor, podrá realizar lo mismo que este pero añadiendo además la gestión de profesores.

Se definirán los casos de uso a través de Diagramas de Casos de Uso y de Tablas de Casos de Uso, donde se definirá las características principal que deseamos que el software sea capaz de realizar.

3.1.1 DIAGRAMAS DE CASOS DE USO

A través de estos diagramas, se podrá ver de una manera más sencilla el funcionamiento global de la aplicación. La figura 3.1 presenta el diagrama de casos de uso que sigue la aplicación.

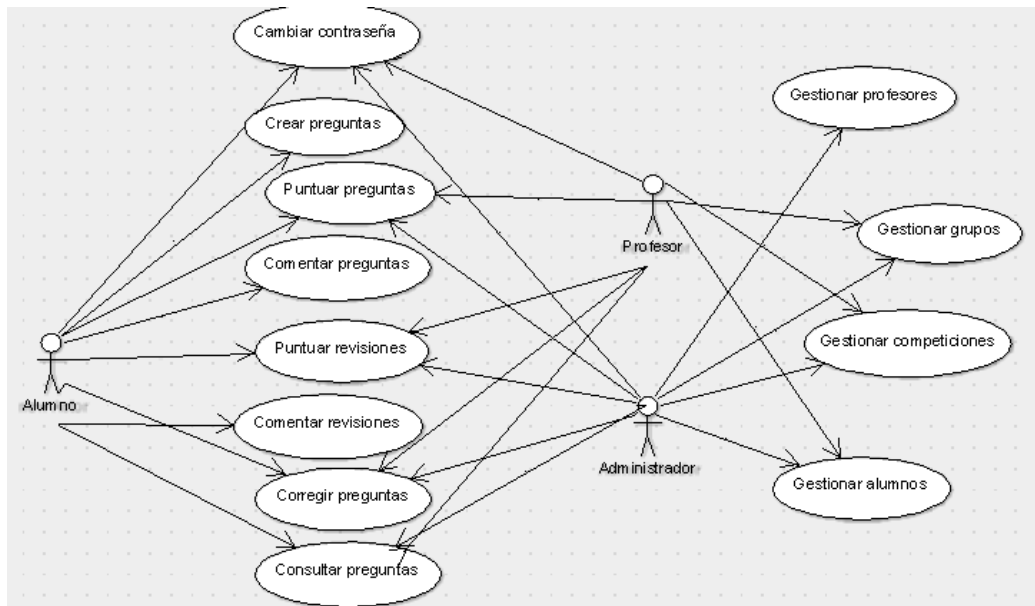


Figura 3.1 Diagrama de Casos de Uso

3.1.2 FUNCIONALIDADES OFRECIDAS

En este apartado se va a ver de forma más detallada las funcionalidades principales y sus requisitos. En concreto, se va a centrar en el cambio de contraseña, la creación de preguntas, puntuación y comentarios de estas, puntuación de revisiones y comentarios de estas, corrección y consulta de preguntas y la gestión de grupos, competencias, alumnos y profesores.

CU001 Cambiar contraseña

Actores. Alumno, profesor, administrador

Descripción. Podrá cambiar de contraseña cuando lo desee. Si es la primera vez que accede al sistema, el cambio de contraseña es necesario.

Precondiciones. Que haya accedido a la aplicación.

Flujo principal. Login. Si es primera vez que accede: cambio de contraseña obligatorio. Si no es primera vez: cambia la contraseña cuando quiera.

Postcondiciones. El usuario ha cambiado la contraseña y la próxima vez que acceda a la aplicación deberá escribir la nueva.

Excepciones. La contraseña no cumple con las especificaciones.

CU002 Crear preguntas

Actores. Alumno

Descripción. El alumno deberá crear preguntas en la fase Escritor del juego

Precondiciones. Que la competición este abierta para jugar la fase escritor.

Flujo principal. Login. Acceder a la competición que se desea que esté disponible.

Postcondiciones. El alumno ha sido capaz de crear las preguntas y pasa a la siguiente fase.

Excepciones. Que no pueda acceder a la fase escritor ya que haya pasado la fecha .

CU003 Puntuar preguntas

Actores. Alumno, profesor, administrador

Descripción. El usuario será capaz de puntuar las preguntas realizadas por el alumno. Si el usuario es un alumno, puntuará la pregunta en la fase de revisor. Si el usuario es un profesor o administrador, lo hará en la fase de corrección del profesor.

Precondiciones. Que la fase anterior se haya realizado correctamente y que la competición se encuentre abierta para jugar la fase correspondiente a la puntuación de preguntas.

Flujo principal. Login. Acceder a la competición a la que se quiera jugar.

Postcondiciones. El usuario ha sido capaz de puntuar las preguntas y pasa a la fase siguiente.

Excepciones. Que la fase se encuentre fuera de fecha y ya no pueda realizarse.

CU004 Comentar preguntas

Actores. Alumno

Descripción. El alumno puede realizar comentarios sobre la pregunta que le ha tocado revisar en la fase revisor y una posible mejora para esta.

Precondiciones. Que los alumnos hayan hecho preguntas y que no le toque revisar su propia pregunta. Haber realizado la fase escritor.

Flujo principal. Login. Acceder a la competición que se quiera jugar.

Postcondiciones. El alumno ha sido capaz de comentar las preguntas y realizar la fase revisor y pase a la siguiente fase.

Excepciones. Que la fase se encuentre fuera de fecha y ya no pueda realizarse.

CU005Puntuar revisiones

Actores. Alumno, profesor, administrador

Descripción. El usuario será capaz de puntuar las revisiones que han hecho de sus preguntas. Si el usuario es un alumno, puntuará la revisión en la fase corrector. Si el usuario es un profesor o administrador, lo hará en la fase de corrección del profesor.

Precondiciones. Que la fase anterior se haya realizado correctamente y que la competición se encuentre abierta para jugar la fase correspondiente a la puntuación de revisiones.

Flujo principal. Login. Acceder a la competición a la que se quiera jugar.

Postcondiciones. El usuario ha sido capaz de puntuar las revisiones y pasa a la fase siguiente.

Excepciones. Que la fase se encuentre fuera de fecha y ya no pueda realizarse.

CU006 Comentar revisiones

Actores. Alumno

Descripción. El alumno puede realizar comentarios sobre la revisión que le han hecho a la pregunta que creó en la fase escritor.

Precondiciones. Que los alumnos completen la fase escritor y se listen sus propias preguntas.

Flujo principal. Login. Acceder a la competición que se quiera jugar.

Postcondiciones. El alumno ha sido capaz de comentar las revisiones y realizar la fase corrector y pase a la siguiente fase.

Excepciones. Que la fase se encuentre fuera de fecha y ya no pueda realizarse.

CU007 Corregir preguntas

Actores. Alumno, profesor, administrador

Descripción. El usuario podrá corregir las preguntas. Si el usuario es un alumno, podrá corregir en la fase corrector las preguntas creadas en la fase escritor. Si es un profesor o administrador, podrá corregir las preguntas del alumno en la fase de corrección de profesor.

Precondiciones. Que los alumnos hayan completado las fases y se listen las preguntas correctamente pertenecientes a cada alumno.

Flujo principal. Login. Acceder a la competición que se quiere jugar.

Postcondiciones. El usuario ha sido capaz de corregir la pregunta deseada y pasa a la siguiente fase.

Excepciones. Que la fase se encuentre fuera de fecha y ya no pueda realizarse.

CU008 Consultar preguntas

Actores. Alumno, profesor, administrador

Descripción. El usuario será capaz de listar las preguntas. Si es un alumno, podrá consultar sus propias preguntas para ver si ha habido corrección por parte del profesor, y acceder a estas para ver el enunciado y sus respuestas. Si es un profesor o administrador, podrá consultar las preguntas del alumno que elija.

Precondiciones. Que la lista de preguntas esté disponible. Para ello, se deben de haber completado las fases necesarias. Si es un alumno, tendrá acceso una vez que la competición haya finalizado. Si es un profesor o administrador, tendrá acceso una vez que el alumno haya hecho la fase corrector.

Flujo principal. Login. Si es alumno acceder a sus preguntas. Si es profesor o administrador, acceder al alumno del cual quiere listar las preguntas.

Postcondiciones. El usuario ha sido capaz de ver las preguntas.

Excepciones. Que no se hayan realizado las fases necesarias o que el alumno no tenga preguntas.

CU009 Gestionar alumnos

Actores. Profesor, administrador

Descripción. El usuario puede ver, modificar, añadir o eliminar alumnos pertenecientes a un grupo.

Precondiciones. Que el alumno pertenezca a un grupo.

Flujo principal. Login. Seleccionar grupo. Si el usuario es un profesor, aparecerá el grupo al que pertenece. Si el usuario es administrador, podrá seleccionar cualquiera de los grupos totales que haya. Listar alumnos y seleccionar alumno o agregar alumno si se quiere añadir uno nuevo.

Postcondiciones. Que el usuario haya sido capaz de modificar, agregar o eliminar al alumno.

Excepciones. Que en el grupo no haya alumnos o que se intente modificar o añadir un alumno existente. No puede haber dos alumnos iguales.

CU010 Gestionar grupos

Actores. Profesor, administrador

Descripción. Si el usuario es un profesor puede ver los grupos y gestionar los alumnos dentro de este. Si el usuario es administrador además podrá modificar, eliminar o añadir grupos nuevos con o sin alumnos.

Precondiciones. Que el grupo exista.

Flujo principal. Login. Si el usuario es un profesor deberá seleccionar el grupo del que quiera ver los alumnos pertenecientes o del grupo en el que quiera añadir alumnos. Si el

usuario es administrado además deberá seleccionar si quiere crear el grupo vacío o añadiendo alumnos mediante CSV e introducir el número del grupo que quiera.

Postcondiciones. Que el usuario sea capaz de ver los alumnos pertenecientes al grupo y poder gestionarlos y en caso de ser administrador además haber creado el grupo deseado.

Excepciones. Que el grupo exista. Que en el grupo haya dos alumnos iguales, en cuyo caso no se podrá añadir.

CU011 Gestionar competiciones

Actores. Profesor, administrador

Descripción. El usuario podrá crear, borrar, ver todas las competiciones, competiciones abiertas, competiciones cerradas, acceder a la fase de corrección y acceder al seguimiento de alumnos, pudiendo ver en qué fase se encuentra cada uno.

Precondiciones. Que el grupo que escoja tenga competiciones disponibles.

Flujo principal. Login. Seleccionar opción que se desee realizar. Seleccionar grupo. Seleccionar competición.

Postcondiciones. Que haya sido capaz de acceder a la competición y haya podido realizar la función deseada.

Excepciones. Que el grupo no tenga competiciones. En caso de realizar la fase de corrección del profesor que no pueda acceder ya que aún no ha llegado la fecha. En caso de seguimiento de alumnos que no haya alumnos que hayan jugado.

CU012 Gestionar profesores

Actores. Administrador

Descripción. El administrador puede crear, eliminar, modificar y listar profesores pertenecientes a un grupo. También puede darle permisos de administrador a un profesor.

Precondiciones. Que el grupo seleccionado para crear, eliminar, modificar o listar profesores no esté vacío.

Flujo principal. Login. Seleccionar opción que se desee hacer. Seleccionar el grupo que se desee. En caso de modificar o eliminar un profesor, seleccionar el profesor que se desee.

Postcondiciones. Que el haya podido realizar la función deseada con los profesores.

Excepción. Que intente crear un profesor que ya se encuentra en el grupo, que el grupo se encuentre vacío en caso de listar los profesores, o que al modificar al profesor ya se encuentre en el grupo.

3.2 DINÁMICA DEL JUEGO

En esta sección se explicará con más detalle de cómo se ha hecho anteriormente, en que consiste el juego. Como se ha comentado anteriormente, este juego consta de 4 fases para el alumno y de una fase para el profesor, por lo que en total presenta 5 fases diferentes. El profesor es el que crea la competición, poniendo las fechas en las que se desarrollará cada fase y el número de preguntas y revisiones que el alumno tendrá que hacer y corregir. A continuación se describen las distintas fases:

- **FASE ESCRITOR:** es la primera fase de la competición y tendrá comienzo en la fecha en la que haya puesto el profesor. En esta fase va a intervenir el alumno, ya que será quien cree las preguntas. El número de preguntas será el número que haya determinado el profesor en el momento en que creó la competición. Las preguntas que cree deben tener 4 posibles opciones de respuesta, siendo una de ellas la correcta. Una vez el alumno cree las preguntas, deberá validarlas y en caso de que quiera cambiarlas, podrá volver a escribirlas. Una vez validadas, pasará a la siguiente fase que empezará en la fecha que el profesor haya indicado.
- **FASE REVISOR:** es la segunda fase de la competición. En esta fase el alumno debe contestar cierto número de preguntas que han salido aleatoriamente y que fueron creadas en la fase escritor por otros alumnos. Una vez contestadas, deberá puntuar la pregunta en función de la dificultad que haya tenido esta. También deberá añadir un comentario sobre su opinión acerca de la pregunta. Y además, deberá escribir una posible mejora de la pregunta para que el escritor la tenga en cuenta en la siguiente fase. Una vez puntuadas y valoradas las preguntas, el alumno pasa a la siguiente fase.
- **FASE CORRECTOR:** es la tercera fase de la competición. En esta fase el alumno recibirá las preguntas que ha creado en la fase escritor junto con las revisiones que hayan hecho los otros alumnos. El alumno se guiará por las revisiones que le hayan hecho, y en caso de querer cambiar la pregunta, podrá crear una nueva. Una vez decidido si corrige o no la pregunta, deberá puntuar las revisiones que han tenido y añadir un comentario sobre la revisión. Podrá comentar el por qué o por qué no ha corregido la pregunta para que el profesor lo tenga en cuenta en la siguiente fase. Una vez hecho esto, se pasará a la siguiente fase, que sería la fase del profesor pero que es transparente para el alumno.
- **FASE CORRECCIÓN PROFESOR:** es la cuarta fase de la competición. En esta fase es el profesor el que entra en juego. El profesor deberá puntuar las preguntas que han hecho los alumnos y las revisiones de las preguntas. Además deberá decidir si la pregunta es válida para pasar a la fase final de la competición. En caso de que sea válida, simplemente deberá puntuar las

preguntas y revisiones. En caso contrario, deberá corregir la pregunta creando una nueva, siendo la que pase a la fase final de la competición. Una vez validadas todas las preguntas, se pasará a la última fase.

- FASE JUEGO: es la quinta y última fase de la competición. En esta fase el alumno solamente deberá contestar las preguntas que le aparezcan de forma aleatoria, verificando que no le toque ninguna que el haya creado. Una vez contestadas todas las preguntas, finalizará la competición.

Una vez acabada la competición, el alumno podrá consultar los resultados obtenidos como escritor, como revisor, como corrector y los fallos y aciertos tanto del juego inicial de la fase revisor, como del juego final de la fase juego. Además podrá ser capaz de consultar las preguntas falladas y acertadas, las preguntas corregidas por el profesor y sus propias preguntas, tanto las iniciales como las corregidas.

El profesor por otro lado, también podrá consultar los resultados obtenidos de cada alumno así como sus las preguntas iniciales, revisiones, preguntas corregidas, comentarios y puntuaciones que han hecho. Además, podrá hacer un seguimiento de los alumnos para saber en qué fase se encuentra cada uno.

3.3 REQUISITOS

En esta sección se van a detallar los requisitos de la aplicación y de los usuarios.

3.3.1 Requisitos de la aplicación

En este apartado se van a mencionar los requisitos que son necesarios para la aplicación y que afecta tanto a los alumnos como al profesor administrador y al profesor no administrador.

- 1- El acceso a la aplicación deberá realizarse, en caso de ser alumno a través de su NIA y una contraseña y, en caso de ser profesor (administrador o no), deberá introducir su login (será un alias) y su contraseña.
- 2- Los usuarios deberá cambiar de contraseña. Si es la primera vez que acceden a la aplicación, es necesario cambiar la contraseña. Por el contrario, si ya han accedido más veces a la aplicación, podrán cambiar la contraseña cuando ellos lo deseen.
- 3- La aplicación deberá controlar el acceso para permitir acceder solamente a los usuarios que se encuentren registrados.
- 4- El registro de alumnos nuevos lo podrán hacer los profesores (administradores o no).
- 5- El registro de profesores nuevos lo podrán hacer los profesores administradores.

3.3.2 Requisitos de los profesores

En este apartado se van a mencionar los requisitos del profesor administrador y no administrador.

3.3.2.1 Grupos

Los profesores administradores y no administradores serán los que puedan realizar acciones con los grupos. Sin embargo, los profesores no administradores tendrán menos acciones que los administradores.

Las acciones que pueden realizar los profesores no administradores con los grupos son:

- 6- Añadir alumnos al grupo manualmente.
- 7- Añadir alumnos al grupo a través de un archivo CSV. Este archivo se cargará desde la aplicación y tiene un formato determinado.
- 8- Editar alumnos pertenecientes al grupo.
- 9- Borrar alumnos del grupo.
- 10- Consultar los alumnos de un grupo.
- 11- Consultar profesores de un grupo.
- 12- Guardar los alumnos de un grupo en el archivo CSV que se le indique.
- 13- Guardar grupos de igual manera que el requisito 12.

Por otro lado, los profesores administradores pueden realizar a parte de estas acciones, unas cuantas más que son:

- 14- Crear un grupo vacío.
- 15- Crear un grupo importando alumnos de un archivo CSV.
- 16- Borrar grupo, borrando también toda la información que haga referencia a dicho grupo (competiciones, alumnos, profesores...).

3.3.2.2 Alumnos

El profesor no administrador deberá realizar las siguientes acciones con los alumnos:

- 17- Crear alumnos nuevos.
- 18- Editar datos de los alumnos.
- 19- Eliminar alumnos y toda la información creada a partir de ellos.
- 20- Consultar preguntas realizadas por los alumnos.
- 21- Consultar revisiones realizadas por los alumnos.
- 22- Consultar correcciones realizadas de los alumnos.
- 23- Consultar preguntas jugadas por los alumnos.
- 24- Consultar resultados de los alumnos.
- 25- Consultar alumnos que han creado preguntas.
- 26- Consultar alumnos que han hecho revisiones
- 27- Consultar alumnos que han hecho correcciones.
- 28- Consultar alumnos que han jugado la fase final.
- 29- Consultar datos de un alumno.
- 30- Consultar la fase que ha realizado cada alumno.
- 31- Corregir preguntas creadas por los alumnos.

3.3.2.3 Competiciones

En este caso, los profesores administradores y no administradores podrán realizar lo mismo:

- 32- Crear competiciones. Se deberá introducir las fechas de inicio y fin de las fases, así como el número de revisiones y preguntas a realizar en cada fase e indicar el grupo donde se realizará dicha competición.
- 33- Borrar competiciones y toda la información que haga referencia a la competición.
- 34- Listar todas las competiciones.
- 35- Listar competiciones abiertas
- 36- Listar competiciones cerradas.
- 37- Consultar estadísticas de la competición seleccionada.
- 38- Exportar preguntas finales de la competición en un archivo CSV.
- 39- Consultar preguntas finales de la competición con los alumnos que hayan jugado dichas preguntas.
- 40- Guardar todos los datos de las competiciones (preguntas, revisiones, correcciones, competiciones, respuestas).
- 41- Jugar la competición seleccionada.

3.3.2.4 Profesores

Para este caso, sólo van a poder realizar acciones los profesores administradores:

- 42- Crear profesor. Podrá crear un profesor que sea o no administrador. En caso de crear un profesor no administrador, tendrá permisos sobre este. Y en caso de crear un profesor administrador, no tendrá permisos sobre este.
- 43- Consultar datos de profesores no administradores de un grupo.
- 44- Editar datos de un profesor no administrador.
- 45- Eliminar a un profesor no administrador, eliminando toda la información que haga referencia a este.

3.3.2.5 Preguntas

Los profesores van a poder realizar las siguientes acciones:

- 46- Corregir preguntas del alumno. Se editará la pregunta, escribiendo un nuevo enunciado y respuestas.
- 47- Consultar preguntas creadas del alumno.
- 48- Consultar preguntas corregidas del alumno.
- 49- Consultar preguntas finales de la fase juego.
- 50- Consultar preguntas acertadas o falladas del alumno en la fase revisor.
- 51- Consultar preguntas acertadas o falladas del alumno en la fase juego.
- 52- Guardar las preguntas que se han creado a lo largo de la competición, con sus posibles respuestas y su respuesta correcta en un archivo CSV.

- 53- Guardar las respuestas que han dado los alumnos a las preguntas que se han creado en la competición en un archivo CSV.
- 54- Puntuar preguntas iniciales (fase escritor).
- 55- Puntuar preguntas finales (fase corrector).

3.3.2.6 Revisiones

En cuanto a las revisiones, lo revisiones deben poder:

- 56- Consultar revisiones hechas por un alumno.
- 57- Consultar revisiones asociadas a preguntas.
- 58- Consultar los comentarios de las revisiones que han hecho los alumnos.
- 59- Guardar las revisiones de una competición en el archivo CSV que se indique.
- 60- Puntuar revisiones.

3.3.3 Requisitos de los alumnos

En este apartado se van a mencionar los requisitos que van a tener los alumnos.

- 61- Consultar competiciones cerradas.
- 62- Consultar todas las competiciones que están abiertas.
- 63- Acceder a las competiciones que estén abiertas.
- 64- Crear preguntas. Deberá escribir el enunciado de la pregunta, con 4 posibles respuestas e indicar la respuesta correcta.
- 65- Corregir preguntas. El alumno podrá editar su pregunta, escribiendo enunciado y respuestas nuevas.
- 66- Consultar preguntas creadas.
- 67- Consultar preguntas corregidas por el mismo alumno.
- 68- Consultar preguntas corregidas por el profesor.
- 69- Consultar resultados como escritor.
- 70- Consultar resultados como revisor.
- 71- Consultar resultados como corrector.
- 72- Consultar fallos y aciertos de las preguntas iniciales de la fase revisor.
- 73- Consultar fallos y aciertos de las preguntas finales de la fase juego.

3.4 MODELO DE DATOS

En este apartado se va a explicar el modelo de datos de la aplicación, que es fundamental ya que se trata de la estructura lógica que seguirán estos, es decir, la forma en la que está organizada la base de datos.

El modelo de datos que se empleara en la aplicación es un modelo relacional, ya que se establecen relaciones entre los campos de tablas diferentes.

Para el desarrollo de la aplicación, se han empleado 14 tablas diferentes, de las cuales 7 pertenecen a las clases que componen dicha aplicación.

A continuación, se muestra el diseño de la base de datos en la figura 3.2

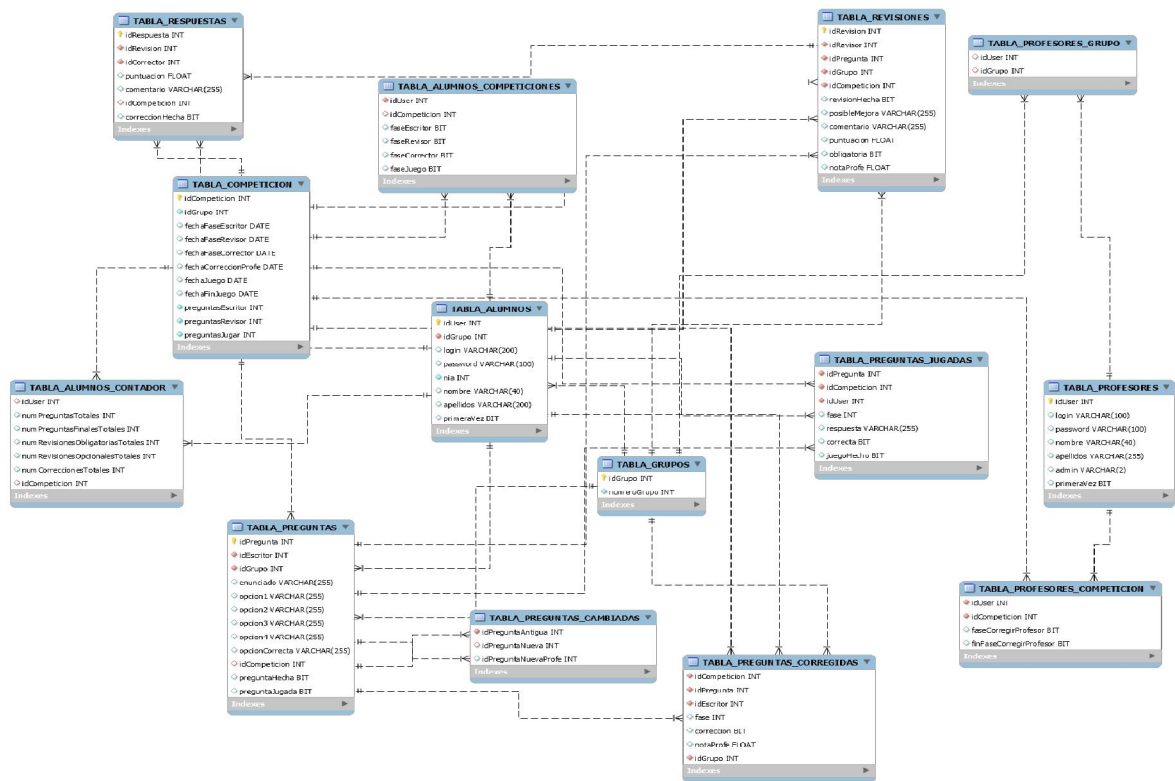


Figura 3.2 Diseño de la base de datos

Como se puede apreciar, hay varias tablas que hacen referencia a preguntas, a alumnos y a profesores. Esto es debido a que son tablas cruzadas, ya que van a requerir campos que relacionen las diferentes tablas.

A continuación se va a ver en detalle cada tabla.

3.4.1 TABLAS

En este apartado se explicará en detalle cada una de las tablas que forma la base de datos.

3.4.1.1 Información sobre el alumno

TABLA_ALUMNOS	
	idUser INT
	idGrupo INT
	login VARCHAR(200)
	password VARCHAR(100)
	nia INT
	nombre VARCHAR(40)
	apellidos VARCHAR(200)
	primeraVez BIT
Indexes	

Figura 3.3 Tabla alumnos

Esta tabla representa a los alumnos de la competición. Dentro se pueden encontrar los siguientes campos:

- IdUser (int): es el identificador del alumno. Este id es único para cada alumno. Es PK de la tabla.
- IdGrupo (int): es el identificador del grupo en el que estará registrado el alumno. Es FK de la tabla tabla_Grupos.
- Login (varchar): es el nombre de usuario con el que el alumno accederá a la aplicación. El login será el mismo que el NIA del alumno.
- Password (varchar): es la contraseña del usuario. Está codificada en MD5. La contraseña por defecto que se le adjudicará al alumno en el momento de su registro será su DNI. Una vez registrado, deberá cambiarla la primera vez que acceda a la aplicación.
- NIA (int): es el NIA del alumno. El NIA también será único para cada alumno.
- Nombre (varchar): es el nombre del alumno.
- Apellidos (varchar): son los apellidos del alumno.
- PrimeraVez (bit): determina si es la primera vez que el alumno accede o no. Si es la primera vez que accede, deberá cambiar la contraseña. Toma el valor 0 cuando es la primera vez, y valor 1 cuando no.

3.4.1.2 Información sobre el alumno y las competiciones



The screenshot shows a database table named 'TABLA_ALUMNOS_COMPETICIONES'. The table has the following fields:

Field Name	Field Type
idUser	INT
idCompeticion	INT
faseEscritor	BIT
faseRevisor	BIT
faseCorrector	BIT
faseJuego	BIT

Below the fields, there is a section labeled 'Indexes' with a right-pointing arrow.

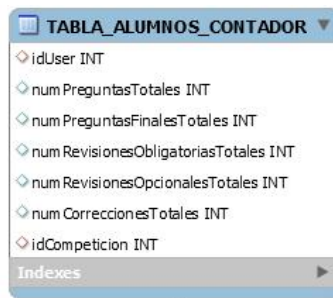
Figura 3.4 Tabla alumnos competiciones

Esta tabla presenta los alumnos y las competiciones que han realizado. Además, guarda la fase en la que se encuentra cada alumno en cada competición. Se encuentran los siguientes campos:

- IdUser (int): es el identificador del alumno. Es único y es FK del idUser de la tabla tabla_Alumnos.
- IdCompeticion (int): es el identificado de la competición. Es único y es FK del idCompeticion de la tabla tabla_Competicion.
- FaseEscritor (bit): determina si el alumno ha realizado la fase escritor. Si toma el valor 0, no ha realizado la fase. So toma el valor 1, si ha realizado la fase.

- FaseRevisor (bit): determina si el alumno ha realizado la fase revisor. Si toma el valor 0, no ha realizado la fase. Si toma el valor 1, si ha realizado la fase.
- FaseCorrector (bit): determina si el alumno ha realizado la fase corrector. Si toma el valor 0, no ha realizado la fase. Si toma el valor 1, si ha realizado la fase.
- FaseJuego (bit): determina si el alumno ha realizado la fase juego. Si toma el valor 0, no ha realizado la fase. Si toma el valor 1, si ha realizado la fase.

3.4.1.3 Información sobre el alumno y el número de preguntas, revisiones y correcciones



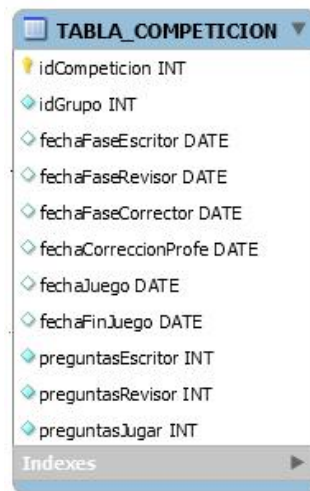
TABLA_ALUMNOS_CONTADOR	
idUser	INT
num PreguntasTotales	INT
num PreguntasFinalesTotales	INT
num RevisionesObligatoriasTotales	INT
num RevisionesOpcionalesTotales	INT
num CorreccionesTotales	INT
idCompeticion	INT
Indexes	

Figura 3.5 Tabla alumnos contador

En esta tabla se almacenará el número de preguntas, revisiones y respuestas que han hecho los alumnos en cada competición. Sus campos son:

- IdUser (int): es el identificador del alumno. Es único y es FK del idUser de la tabla tabla_Alumnos.
- NumPreguntasTotales (int): es el número de preguntas que el alumno ha realizado en la fase de escritor.
- NumRevisionesObligatoriasTotales (int): es el número de revisiones obligatorias que el alumno ha hecho en la fase revisor. Estas revisiones son las que el profesor determinó realizar en el momento de la creación de la competición.
- NumRevisionesOpcionalesTotales (int): es el número de revisiones opcionales que el alumno indicó hacer en la fase revisor.
- NumCorreccionesTotales (int): es el número de correcciones que el alumno ha realizado en la fase de corrector.
- IdCompeticion (int): es el identificador de la competición. Es único y es FK del idCompeticion de la tabla tabla_Competicion.

3.4.1.4 Información sobre la competición



TABLA_COMPETICION	
idCompeticion	INT
idGrupo	INT
fechaFaseEscrivor	DATE
fechaFaseRevisor	DATE
fechaFaseCorrector	DATE
fechaCorreccionProfe	DATE
fechaJuego	DATE
fechaFinJuego	DATE
preguntasEscrivor	INT
preguntasRevisor	INT
preguntasJugar	INT

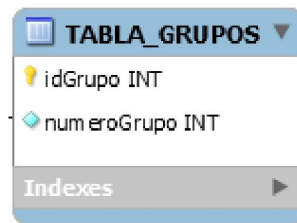
Indexes

Figura 3.6 Tabla competición

En esta tabla se guardarán los datos de las competiciones. Sus campos son:

- IdCompeticion (int): es el identificador de la competición. Este id es único para cada competición. Es PK de la tabla.
- IdGrupo (int): es el identificador del grupo al que pertenece la competición. Es único y es FK del idGrupo de la tabla tabla_Grupos.
- FechaFaseEscrivor (date): es la fecha en la que tendrá comienzo la fase escritor de la competición, es decir, la fecha en la comenzará la competición.
- FechaFaseRevisor (date): es la fecha en la que tendrá comienzo la fase revisor de la competición y por lo tanto, la fecha en la que finaliza la fase escritor.
- FechaFaseCorrector (date): es la fecha en la que tendrá comienzo la fase corrector de la competición y por lo tanto, la fecha en la que finaliza la fase revisor.
- FechaCorrecciónProfesor (date): es la fecha en la que tendrá comienzo la fase de corrección del profesor, y por lo tanto, la fecha en la que finaliza la fase corrector.
- FechaJuego (date): es la fecha en la que tendrá comienzo la fase de juego, y por lo tanto, la fecha en la que finaliza la fase de corrección del profesor.
- FechaFinJuego (date): es la fecha en la que finalizará la competición. Una vez finalizada la competición, no se podrá acceder a ella ya que estará cerrada.
- PreguntasEscrivor (int): es el número de preguntas que el alumno deberá crear en la fase escritor. Además, hace referencia a las preguntas que tendrá que corregir en la fase corrector, ya que serán las preguntas que ha creado inicialmente.
- PreguntasRevisor (int): es el número de preguntas que el alumno deberá revisar en la fase revisor.
- PreguntasJugar (int): es el número de preguntas que el alumno deberá contestar en la fase juego.

3.4.1.5 Información sobre los grupos



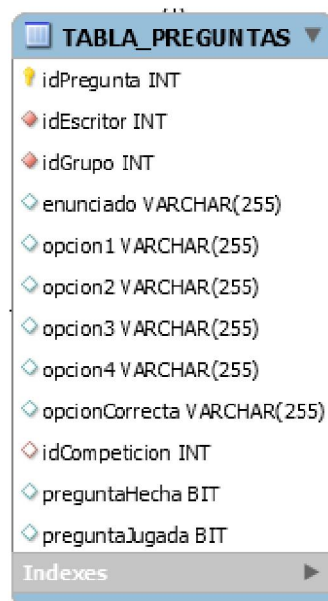
TABLA_GRUPOS	
idGrupo	INT
numeroGrupo	INT
Indexes	

Figura 3.7 Tabla grupos

En esta tabla se guardarán los grupos que el profesor vaya creando. Sus campos son:

- IdGrupo (int): es el identificador del grupo. Este id es único y es PK de la tabla.
- NumeroGrupo (int): es el número que va a tener el grupo. Con este número es como se identificará fuera de la base de datos.

3.4.1.6 Información sobre las preguntas



TABLA_PREGUNTAS	
idPregunta	INT
idEscritor	INT
idGrupo	INT
enunciado	VARCHAR(255)
opcion1	VARCHAR(255)
opcion2	VARCHAR(255)
opcion3	VARCHAR(255)
opcion4	VARCHAR(255)
opcionCorrecta	VARCHAR(255)
idCompeticion	INT
preguntaHecha	BIT
preguntaJugada	BIT
Indexes	

Figura 3.8 Tabla preguntas

En esta tabla se guardarán todas las preguntas que se creen, tanto si se crean por primera vez, como si se corrigen. Sus campos son:

- IdPregunta (int): es el identificador de la pregunta. Este id es único y es PK de la tabla.
- IdEscritor (int): es el identificador del alumno que ha creado la pregunta. Es único y es FK del idUser de la tabla tabla_Alumnos.
- IdGrupo (int): es el identificador del grupo al que pertenece la pregunta. Es único y es FK del idGrupo de la tabla tabla_Grupos.

- Enunciado (varchar): es el enunciado de la pregunta
- Opcion1 (varchar): es la primera opción que se le pone a la pregunta como posible respuesta.
- Opcion2 (varchar): es la segunda opción que se le pone a la pregunta como posible respuesta.
- Opcion3 (varchar): es la tercera opción que se le pone a la pregunta como posible respuesta.
- Opcion4 (varchar): es la tercera opción que se le pone a la pregunta como posible respuesta.
- OpcionCorrecta (varchar): es la opción correcta de entre las 4 opciones anteriores. Es la respuesta de la pregunta.
- IdCompeticion (int): es el identificador de la competición a la que pertenece la pregunta. Es único y es FK del idCompeticion de la tabla tabla_Competicion.
- PreguntaHecha (bit): es un campo opcional que sirve para acotar las búsquedas de preguntas y facilitar su acceso en determinados caso. Cuando la pregunta es creada, toma el valor 1.
- PreguntaJugada (bit): determinada si la pregunta ha sido usada en la fase juego como pregunta a responder. Es un campo opcional que sirve para acotar las búsquedas de preguntas y facilitar su acceso en determinados caso. Si la pregunta no ha sido usada, toma el valor 0. Si ha sido usada en la fase juego, toma el valor 1.

3.4.1.7 Información sobre las preguntas actualizadas



Figura 3.9 Tabla preguntas cambiadas

Esta tabla se va a usar para guardar la relación que existe entre la pregunta creada inicialmente y la modificada, es decir, si el alumno en la fase corrector o el profesor en la fase de corrección del profesor corrige dicha pregunta, la pregunta se guardará con un id nuevo pero conservaremos el id de la pregunta que se ha corregido para luego hacer consultas. Sus campos son:

- IdPregunta (int): es el identificador de la pregunta sin corregir, es decir, la inicial. Es FK del idPregunta de la tabla tabla_Preguntas.
- IdPreguntaNueva (int): es el identificador de la pregunta nueva al hacer la corrección por parte del alumno. Es FK del idPregunta de la tabla tabla_Preguntas.
- IdPreguntaNuevaProfe (int): es el identificador de la pregunta nueva al hacer la corrección por parte del profesor. Es FK del idPregunta de la tabla tabla_Preguntas.

3.4.1.8 Información sobre las preguntas jugadas



The screenshot shows a window titled 'TABLA_PREGUNTAS_JUGADAS'. It lists the following fields with their data types and constraints:

Field Name	Data Type	Constraint
idPregunta	INT	Primary Key (red diamond)
idCompeticion	INT	Foreign Key (red diamond)
idUser	INT	Foreign Key (red diamond)
fase	INT	None (green diamond)
respuesta	VARCHAR(255)	None (green diamond)
correcta	BIT	None (green diamond)
juegoHecho	BIT	None (green diamond)

At the bottom, there is a tab labeled 'Indexes' with a right-pointing arrow.

Figura 3.10 Tabla preguntas jugadas

Esta tabla guardará las respuestas que el alumno indique a la hora de contestar las preguntas. Además, guardará la fase a la que pertenece la pregunta, si es inicial de la fase revisor, o si es final de la fase juego y si ha sido correcta o no. Presenta los siguientes campos:

- IdPregunta (int): es el identificador de la pregunta que se ha contestado. Es FK del idPregunta de la tabla tabla_Preguntas.
- IdCompeticion (int): es el identificador de la competición en la que se encuentra la pregunta que se ha contestado. Es FK del idCompeticion de la tabla tabla_Competicion.
- IdUser (int): es el identificador del alumno que ha contestado la pregunta. Es FK del idUser de la tabla tabla_Alumnos.
- Fase (int): determina la fase en la que se ha contestado la pregunta. Toma el valor 1 si se ha contestado en la fase revisor y el valor 2 si se ha contestado en la fase juego.
- Respuesta (varchar): es la respuesta que el alumno eligió como opción correcta a la pregunta.
- Correcta (bit): determina si la respuesta que el alumno eligió es correcta (1) o no (0).
- juegoHecho (bit): toma el valor 1 cuando el alumno ya ha realizado el juego de la fase juego.

3.4.1.9 Información sobre las preguntas corregidas



Figura 3.11 Tabla preguntas corregidas

En esta tabla se van a guardar los datos de las preguntas que se han corregido y cuando se han corregido. Sus campos son:

- IdCompeticion (int): es el identificador de la competición en la que se encuentra la pregunta que se ha corregido. Es FK del idCompeticion de la tabla tabla_Competicion.
- IdPregunta (int): es el identificador de la pregunta que se ha corregido. Es FK del idPregunta de la tabla tabla_Preguntas.
- IdEscrivor (int): es el identificador del alumno que ha corregido la pregunta. Es FK del idUser de la tabla tabla_Alumnos.
- Fase (int): determina la fase en la que se encuentra la pregunta. Esto es, si la pregunta es la que se ha creado en la fase escritor, la fase será 1. Si la pregunta está en la fase de corrector, la fase será 2. Y cuando el profesor haya validado y/o corregido la pregunta, la fase será 3.
- Correccion (bit): determina si la pregunta ha sido corregida o no, ya sea por el alumno o por el profesor.
- NotaProfe (float): es la nota que el profesor da en el momento de su corrección en la fase de corrección del profesor.
- IdGrupo (int): es el identificador del grupo al que pertenece la pregunta. Es FK del idGrupo de la tabla tabla_Grupos.

3.4.1.10 Información sobre los profesores

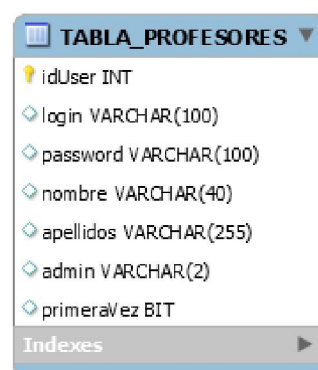


Figura 3.12 Tabla profesores

Esta tabla guarda los datos de los profesores que hayan creado y si son administradores o no. Los campos de esta tabla son:

- IdUser (int): es el identificador del profesor. Este id será único para cada profesor.
- Login (varchar): es el nombre de usuario del profesor con el que podrá acceder a la aplicación.
- Password (varchar): es la contraseña del usuario. Está codificada en MD5. La contraseña por defecto que se le adjudicará al profesor en el momento de su registro será su DNI. Una vez registrado, deberá cambiarla la primera vez que acceda a la aplicación.
- Nombre (varchar): es el nombre del profesor.
- Apellidos (varchar): son los apellidos del profesor.
- Admin (varchar): determina si el profesor es administrador o no. En caso de que sea administrador, podrá acceder a todos los grupos. En caso de que no lo sea, solamente podrá acceder a aquel o aquellos que el admin le haya adjudicado.
- PrimeraVez (bit): determina si es la primera vez que el profesor accede o no a la aplicación. En caso de que sea la primera vez, deberá cambiar la contraseña. Si accede por primera vez, toma el valor 0. Si no accede por primera vez, toma el valor 1.

3.4.1.11 Información sobre los profesores y las competencias

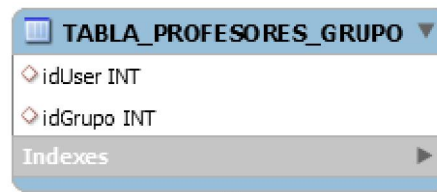


Figura 3.13 Tabla profesores competición

Esta tabla tiene la misma función que la tabla tabla_Alumnos_Competiciones. Se guardará el profesor, la competición que está jugando y si ya ha realizado la fase de su corrección. Sus campos son:

- IdUser (int): es el identificador del profesor. Es FK del idUser de la tabla tabla_Profesores.
- IdCompeticion (int): es el identificador de la competición en la que está realizando su fase. Es FK del idCompeticion de la tabla tabla_Competicion.
- FaseCorregirProfesor (bit): determina si el profesor se encuentra realizando la fase de corrección del profesor.
- FinFaseCorregirProfesor (bit): determina si el profesor ha terminado de realizar su fase.

3.4.1.12 Información sobre los profesores y los grupos



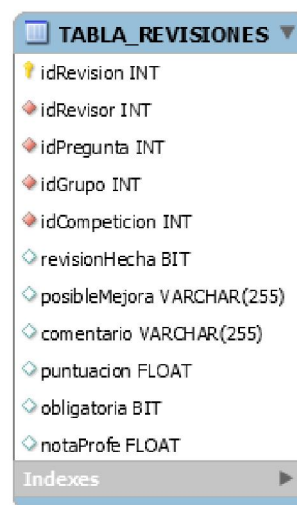
TABLA_PROFESORES_GRUPO	
idUser INT	
idGrupo INT	
Indexes	

Figura 3.14 Tabla profesores grupo

En esta tabla se guardarán los grupos del profesor en caso de que tenga más de uno adjudicado. Sus campos son:

- IdUser (int): es el identificador del profesor. Es FK del idUser de la tabla tabla_Profesores.
- IdGrupo (int): es el identificado del grupo al que pertenece el profesor. Puede tener asociado más de uno ya que puede ser profesor de más de un grupo. Es FK del idGrupo de la tabla tabla_Grupos.

3.4.1.13 Información sobre las revisiones



TABLA_REVISIONES										
idRevision INT										
idRevisor INT										
idPregunta INT										
idGrupo INT										
idCompeticion INT										
revisionHecha BIT										
posibleMejora VARCHAR(255)										
comentario VARCHAR(255)										
puntuacion FLOAT										
obligatoria BIT										
notaProfe FLOAT										
Indexes										

Figura 3.15 Tabla revisiones

En esta tabla se guardarán los datos de las revisiones que se realicen. Sus campos son:

- IdRevision (int): es el identificador que tendrá la revisión. Este id es único para cada revisión. Es PK de la tabla.
- IdRevisor (int): es el identificador del alumno que ha hecho la revisión. Es FK del idUser de la tabla tabla_Alumnos.
- IdPregunta (int): es el identificador de la pregunta que se ha revisado. Es FK del idPregunta de la tabla tabla_Preguntas.
- IdGrupo (int): es el identificador del grupo al que pertenece la revisión. Es FK del idGrupo de la tabla tabla_Grupos.

- IdCompeticion (int): es el identificador de la competición a la que pertenece la revisión. Es FK del idCompeticion de la tabla tabla_Competicion.
- RevisionHecha (bit): es un campo opcional que sirve para acotar las búsquedas de revisiones y facilitar su acceso en determinados caso. Cuando se hace la revisión, toma el valor 1.
- PosibleMejora (varchar): se trata de la posible mejora que el revisor puede dar a la pregunta en el momento de la revisión, para que el escritor de esta la tenga en cuenta en la fase de corrector.
- Comentario (varchar): se trata del comentario que el revisor puede dar de la pregunta, comentando cualquier aspecto de esta, como la dificultad, si está bien creada, etc.
- Puntuacion (float): es la puntuación que el revisor da a cada pregunta.
- Obligatoria (bit): determina si la revisión que se ha hecho es obligatoria. En caso de no ser obligatoria (0), la revisión que se ha hecho es opcional.
- NotaProfe (float): es la nota que el profesor da a la revisión en el momento de la fase de corrección del profesor.

3.4.1.14 Información sobre las correcciones



TABLA_RESPUESTAS	
idRespuesta	INT
idRevision	INT
idCorrector	INT
puntuacion	FLOAT
comentario	VARCHAR(255)
idCompeticion	INT
correccionHecha	BIT
Indexes	

Figura 3.16 Tabla respuestas

En esta tabla se guardarán las respuestas que el alumno da a las revisiones en la fase corrector. Sus campos son:

- IdRespuesta (int): es el identificador que tendrá la corrección. Este id es único para cada corrección. Es PK de la tabla.
- IdRevisión (int): es el identificador de la revisión que se ha corregido. Es FK del idRevision de la tabla tabla_Revisiones.
- IdCompetición (int): es el identificador de la competición en la que se ha creado la corrección. Es FK del idCompeticion de la tabla tabla_Competicion.
- IdCorrector (int): es el identificador del alumno que ha hecho la corrección de la revisión. Es FK del idUser de la tabla tabla_Alumnos.
- Puntuacion (float): es la puntuación que el alumno corrector ha dado a la revisión.

- Comentario (varchar): es el comentario que el alumno corrector puede dar a la revisión. En este comentario puede dar su opinión sobre por qué decide o no corregir la pregunta, si la revisión está bien hecha, etc.
- CorreccionHecha: es un campo opcional que sirve para acotar las búsquedas de correcciones y facilitar su acceso en determinados caso. Cuando se hace la corrección, toma el valor 1.

3.5 MODELOS

Para la realización de la aplicación, se han utilizado las siguientes clases con sus respectivos atributos:

- Alumno. Esta clase hace referencia a la información que el usuario alumno va a tener. Los atributos de esta clase se han explicado en los apartados 3.4.1.1, 3.4.1.2 y 3.4.1.3 a excepción de:
 - DNI: es el DNI del alumno. Será la contraseña por defecto que se le adjudicará al alumno en el momento de su registro. Una vez registrado, deberá cambiarla la primera vez que acceda a la aplicación.
- Profesor. Esta clase hace referencia a la información que el usuario profesor va a tener. Sus atributos han sido definidos en los apartados 3.4.1.10, 3.4.1.11 y 3.4.1.12 a excepción del siguiente atributo:
 - DNI: es el DNI del profesor. Será la contraseña por defecto que se le adjudicará al profesor en el momento de su registro. Una vez registrado, deberá cambiarla la primera vez que acceda a la aplicación.
- Pregunta. Esta clase hace referencia a la información que presentará cada pregunta. Los atributos de dicha clase se han definido en los apartados 3.4.1.6, 3.4.1.7, 3.4.1.8 y 3.4.1.9 a excepción del siguiente atributo:
 - HayCorreccion: determina si la pregunta ha sido corregida o no, ya sea por el alumno o por el profesor.
- Revisión. Esta clase hace referencia a la información que presentará cada revisión. Todos sus atributos han sido definidos en el apartado 3.4.1.13.
- Respuesta. Esta clase hace referencia a la información que presentará cada respuesta de la revisión, es decir, las correcciones que se hagan en la fase corrector. Todos sus atributos han sido definidos en el apartado 3.4.1.14.
- Grupo. Esta clase hace referencia a la información que presentará cada grupo. Sus atributos se han definido en el apartado 3.4.1.5.
- Competición. Esta clase hace referencia a la información que presentará cada competición. Todos sus atributos se han definido en el apartado 3.4.1.4.

3.5.1 DIAGRAMA ENTIDAD-RELACIÓN

En esta sección representaremos un modelo entidad-relación de los datos que se han empleado en la aplicación. Un diagrama o modelo entidad-relación permite representar la estructura de datos de una aplicación, las entidades que tiene, las relaciones entre ellas y las propiedades de cada una de ellas. En la figura 3.17 se puede observar la relación de las clases existentes entre ellas.

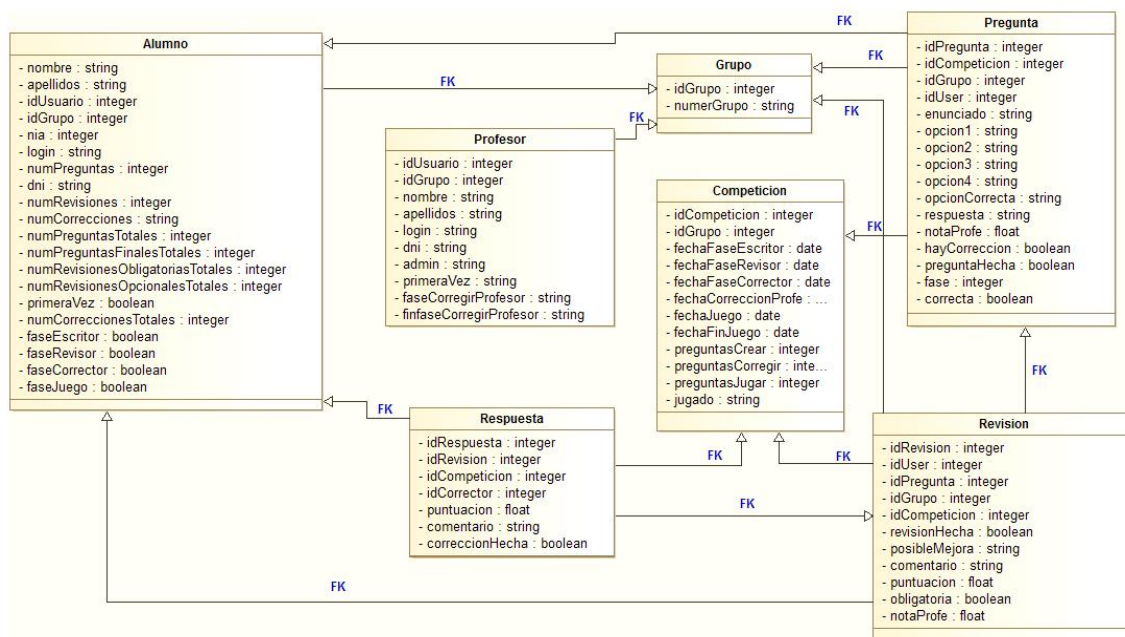


Figura 3.17 Relación de clases

Como se puede observar en la figura 3.17, las revisiones, preguntas, y respuestas tienen relación directa con el alumno, el grupo y la competición, ya que estos van a determinar quién las ha realizado y a qué competición pertenecen. El alumno y profesor tendrán relación directa con el grupo, ya que pertenecen a uno y en caso de que el profesor sea administrador, a más de uno.

3.6 DIAGRAMAS DE FLUJO DEL FUNCIONAMIENTO

En esta sección se va a explicar el recorrido que sigue la aplicación una vez que el usuario (alumno o profesor) la ejecuta a través de diagramas de flujo.

3.6.1 Inicio de la aplicación

El primer usuario que deberá iniciar la aplicación será el administrador de esta, ya que es el que estará registrado en la base de datos y el que debe registrar a los alumnos y a los profesores en caso de que cree a más.



Figura 3.18 Diagrama de flujo inicio aplicación

3.6.2 Inicio sesión del usuario

En primer lugar, el usuario que debe iniciar sesión es el administrador. Para ello, debe introducir su usuario y contraseña. El paso siguiente es realizar una consulta en la BBDD para saber si se encuentran el usuario y la contraseña que se han introducido. En caso de que estén, se hará otra consulta en la que se obtendrá el nombre del administrador a través del login que ha introducido y una vez obtenido, se accederá al menú del administrador donde aparecerá el nombre buscado. En caso de que no tengan coincidencia la contraseña y el usuario introducidos, tendrá que volver a introducir datos que sean válidos.

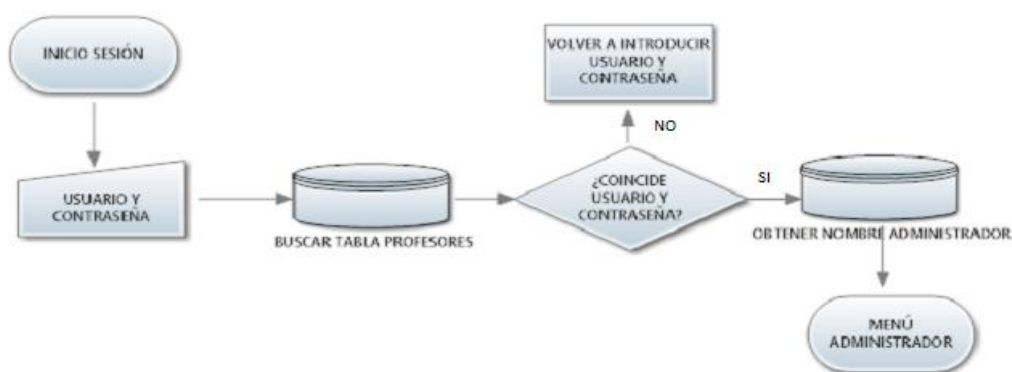


Figura 3.19 Inicio sesión administrador.

3.6.3 Creación de grupos y registro de alumnos y/o profesores

Una vez que el administrador ha accedido a la aplicación, deberá registrar a los alumnos. Para ello, tiene dos formas de hacerlo:

1. Puede crear primero el grupo vacío y luego añadir los alumnos a través de un archivo CSV o manualmente. Para ello creará el grupo asegurándose de que dicho grupo no existe y una vez creado se guardará en la BBDD. En la figura 3.20 se presenta este proceso.

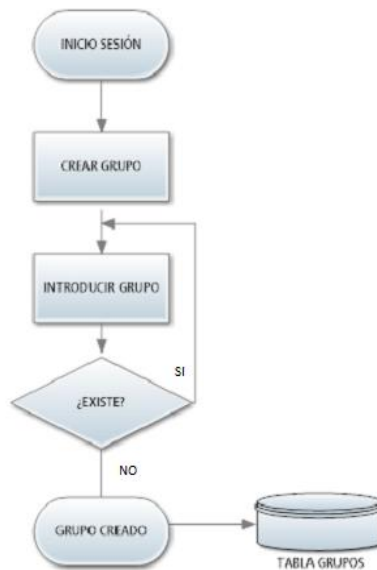


Figura 3.20 Creación de un grupo.

Hecho esto, si quiere añadir alumnos manualmente, deberá elegir primero el grupo que desea editar y luego añadir los alumnos introduciendo su nombre, apellidos, NIA y contraseña. Si quiere añadir alumnos a través de CSV, deberá introducir el archivo CSV con la lista de alumnos y elegir el grupo al que pertenecen estos alumnos. En la figura 3.21, se representa el proceso del registro de alumnos según la forma en la que se desee.

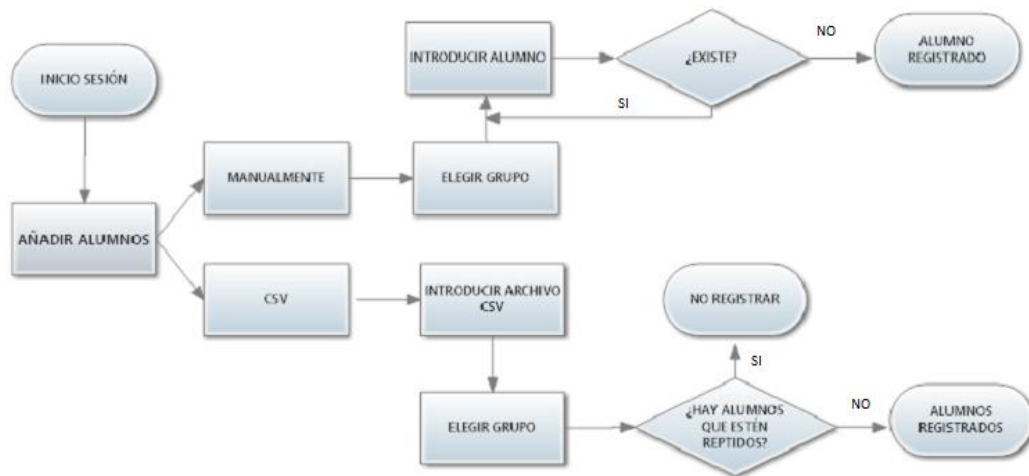


Figura 3.21 Registro de alumnos

El archivo CSV tendrá una lista con los nombres, apellidos, NIA y DNI de los alumnos (que serán sus contraseñas iniciales) y estarán separados por ";", de manera que hace más fácil el poder acceder a los datos. En la figura 3.22 se presenta un ejemplo del archivo CSV.

92.csv	
1	NIU;DNI;APELLIDOS;NOMBRE
2	100274820;12345678F;VERA VILLAJULCA;GRECIA ISABEL

Figura 3.22 Ejemplo de archivo CSV.

En el caso de los profesores, sólo tiene la opción de crearlos manualmente. En el momento de su creación, el administrador podrá especificar si será únicamente profesor (si sólo tiene acceso a su grupo) o si también pasa a ser administrador (tiene acceso a todos los grupos) e indicará el grupo al que pertenece en el caso de ser profesor.

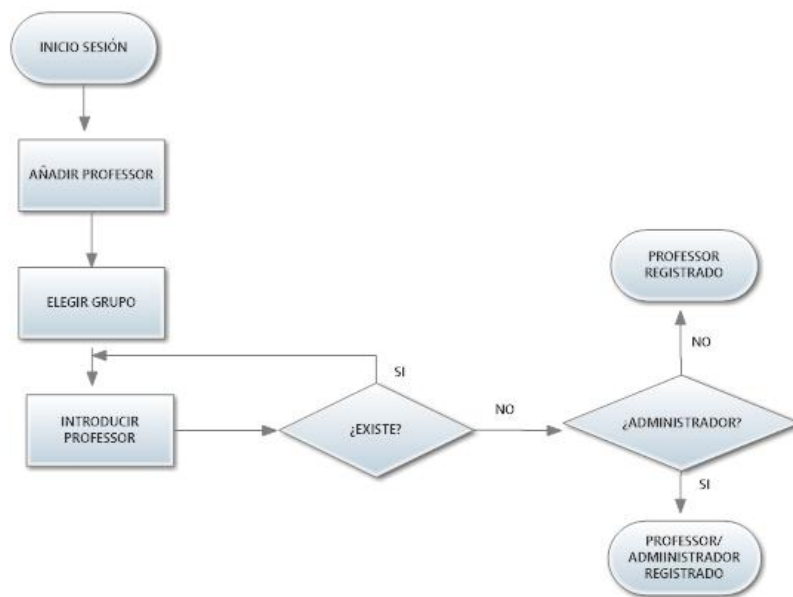


Figura 3.23 Registro de profesores

2. Puede crear el grupo añadiendo directamente los alumnos a través de un archivo CSV. Para ello, deberá introducir el archivo que desee para cargar a los alumnos y luego introducir el grupo que quiere crear, asegurándose de que no exista. En la figura 3.24 se puede observar dicho proceso.

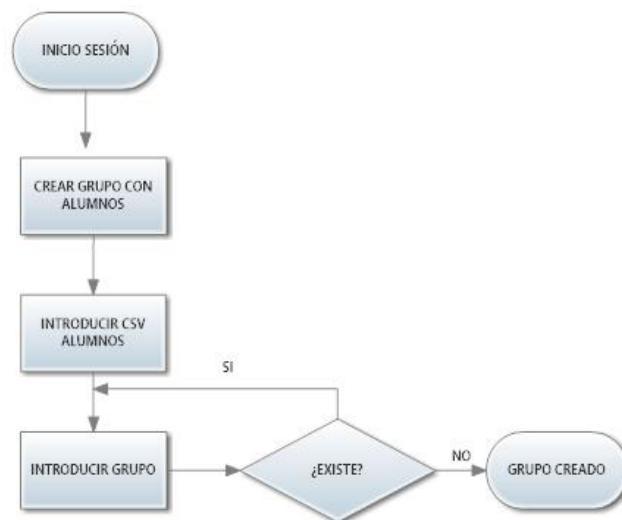


Figura 3.24 Creación grupo a partir de archivo CSV

Una vez realizado uno de los dos pasos descritos anteriormente, los alumnos podrán iniciar sesión.

3.6.4 Inicio sesión alumnos

Una vez registrados los alumnos por el administrador, podrán ser capaces de acceder a la aplicación. Para ello deberán introducir su NIA como usuario y su DNI como contraseña. Si están en la base de datos y es la primera vez que acceden, tendrán que cambiar la contraseña obligatoriamente para poder acceder a su menú. En caso de que no sea la primera vez, podrán acceder al menú sin problemas. La figura 3.25 presenta el proceso de inicio de sesión del alumno.

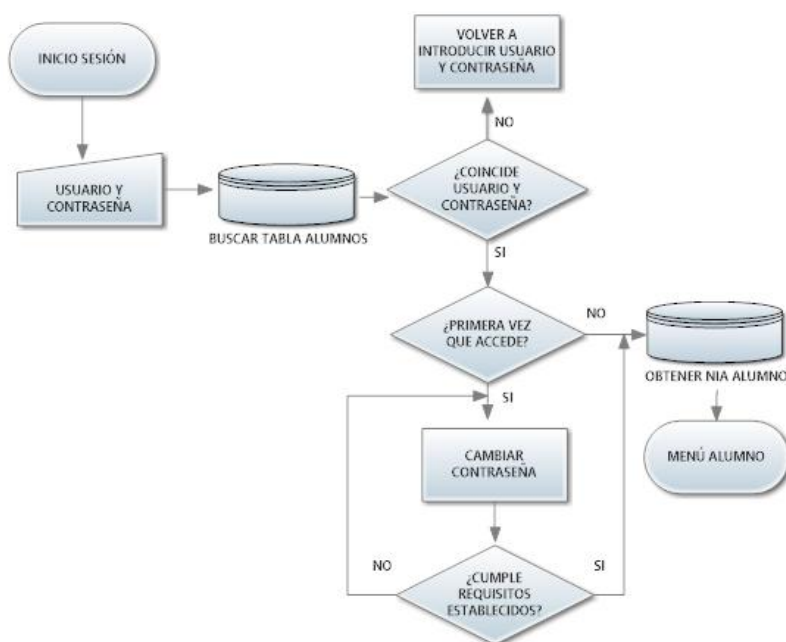


Figura 3.25 Inicio sesión alumno

3.6.5 Menú profesor

El menú que tendrá el profesor será distinto al del alumno, ya que este posee otro tipo de permisos y deberá realizar acciones diferentes al del alumno. En caso de que el profesor sea administrador, contará con otro menú semejante al del profesor normal pero añadiéndole más funcionalidad. A continuación se explica en detalle cada uno de los menús:

- El menú del profesor que no es administrador contará con cuatro opciones a elegir: gestionar grupo, gestionar competiciones, cambiar contraseña o guardar datos de competición.
- Gestionar grupos: dentro de esta opción existen otras dos opciones:
 - Ver grupos: se puede consultar los grupos que hay y listar los alumnos de cada grupo.
 - Editar grupos: esta última opción comprende:
 - Añadir alumnos a partir de un archivo CSV: añadirá alumnos a partir importando un CSV.
 - Añadir alumnos manualmente: añadirá alumnos de forma manual, completando sus campos.

- Modificar alumnos: consiste en poder editar al alumno, ya sea su nombre, apellidos o cambiarle de grupo, y también en poder eliminar al alumno que se escoja. En la figura 3.26 se presenta el diagrama de flujo de gestionar grupos.

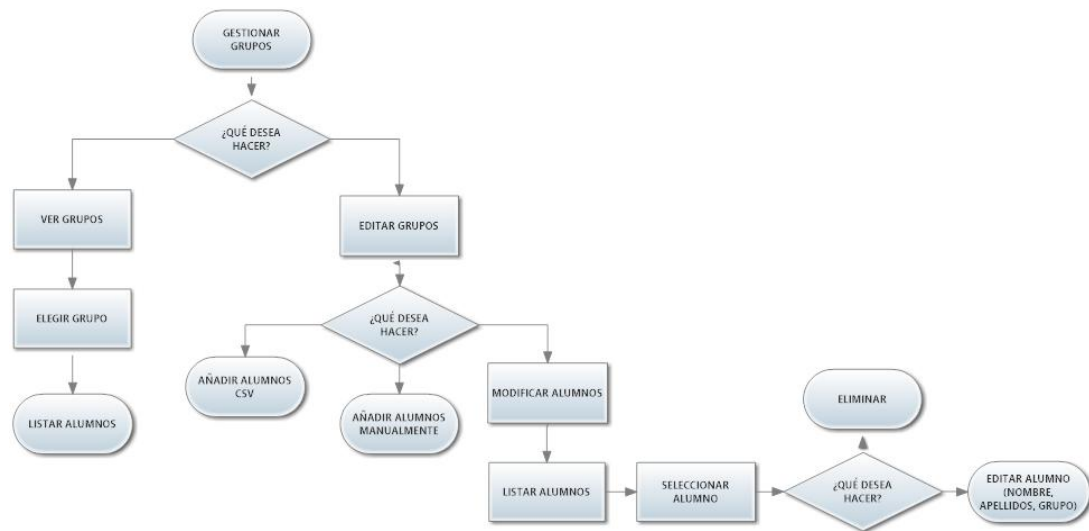


Figura 3.26 Gestionar grupos

- Gestionar competiciones: dentro de esta opción existen las opciones que se muestran a continuación:
 - Todas las competiciones: esta opción listará todas las competiciones creadas, ya estén cerradas o abiertas del grupo escogido.
 - Competiciones abiertas: esta opción listará solamente las competiciones que estén abiertas del grupo escogido, es decir, aquellas en las que la fecha del inicio de la primera fase sea igual o anterior a la actual.
 - Competiciones cerradas: esta opción listará solamente las competiciones del grupo escogido que estén cerradas, es decir, aquellas en las que la fecha de fin de juego sea anterior a la fecha actual.
 - Corregir preguntas finales alumno: esta opción hace referencia a la fase de corrección del profesor. Se escogerá el grupo del cual se quieran listar las competiciones y una vez listadas estas, se escogerá aquella en la que toque realizar dicha fase. Una vez finalizada la corrección, ya no se podrá volver a acceder a corregir las preguntas.
 - Seguimiento de alumnos: esta opción permite que el profesor acceda a todo lo que el alumno ha ido haciendo, se divide en:
 - Alumnos que han creado preguntas: el profesor podrá consultar los alumnos que han hecho preguntas en la fase escritor y si las han cambiado o no en la fase corrector y consultar en detalle las preguntas (enunciado y respuestas).

- Alumnos que han hecho revisiones: el profesor podrá consultar los alumnos que han hecho revisiones en la fase revisor. Se listarán las preguntas pertenecientes a dichas revisiones y se podrá consultar la revisión perteneciente a esa pregunta.
- Alumnos que han hecho correcciones: el profesor podrá consultar los alumnos que han corregido las preguntas iniciales y ver en detalle estas.
- Alumnos que han jugado: el profesor podrá consultar los alumnos que han realizado la fase juego. Se listarán los alumnos y los fallos y aciertos que han tenido en la fase juego. También podrá consultar las preguntas que han fallado y acertado en detalle para que pueda saber que preguntas son las que tiene mejores resultados de acierto.
- Fase que ha realizado cada alumno: el profesor podrá consultar la fase en la que se encuentra cada alumno.
- Resultados de alumnos: el profesor podrá consultar los resultados que el alumno ha obtenido en cada fase.
- Estadísticas competición: el profesor podrá consultar las preguntas que se han jugado en la fase juego y ver la cantidad de alumnos que han acertado y fallado dichas preguntas. Además, podrá acceder a dicho número y consultar los alumnos.
- Crear competición: el profesor tendrá la opción de crear una nueva competición para el grupo que desee.
- Borrar competición: el profesor podrá borrar una competición del grupo que desee.

En la figura 3.27 se ha representado las distintas opciones que hay:

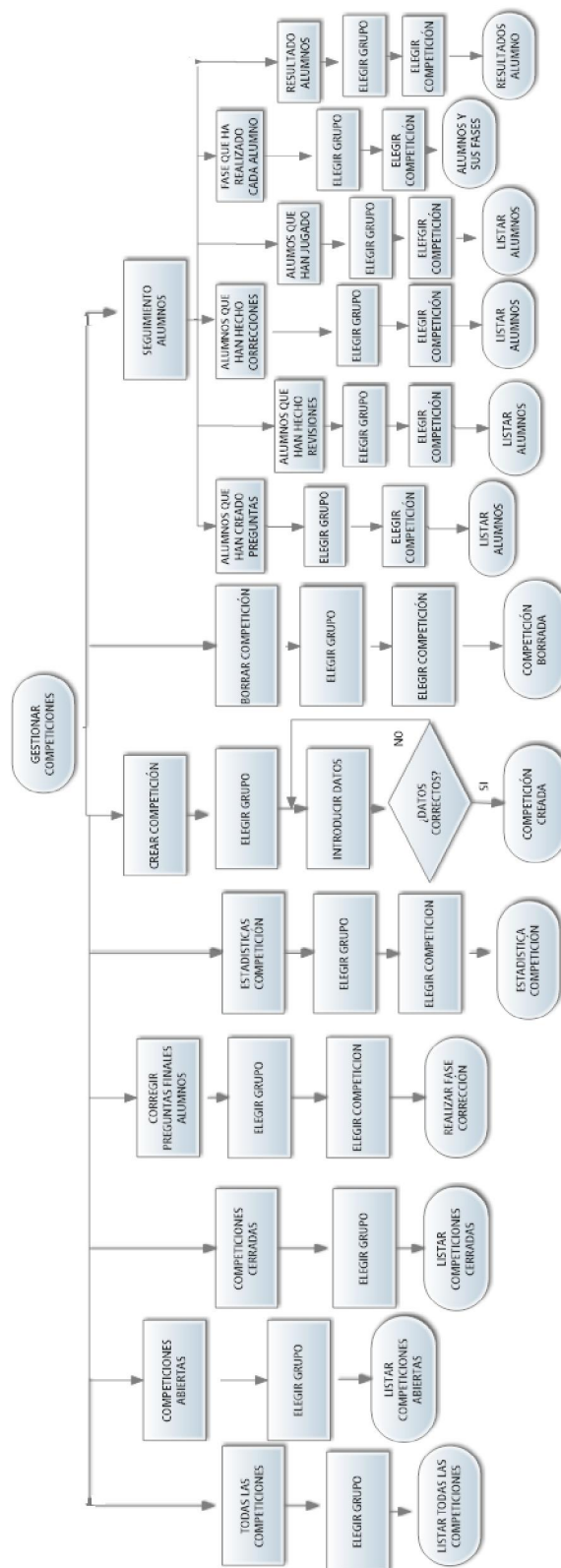


Figura 3.27 Gestionar competiciones

- Cambiar contraseña: el profesor podrá cambiar de contraseña siempre y cuando lo desee. Dicha contraseña deberá cumplir con algunos requisitos específicos para darle más seguridad. La figura 3.28 representa dicho proceso.

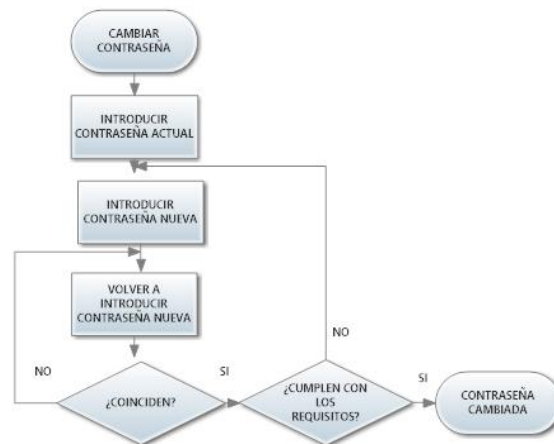


Figura 3.28 Cambiar contraseña profesor

- Guardar datos competición: el profesor podrá guardar los datos de todo el juego cuando desee. Dentro de esta opción, se encuentran varias opciones para que elija lo que desee guardar. Estas son: guardar preguntas, guardar competiciones, guardar alumnos, guardar revisiones y guardar respuestas corrector. Una vez que seleccione que es lo que desea guardar, deberá introducir el directorio con el nombre de la carpeta que donde quiere guardar los datos. En caso de que el archivo ya exista, se encargará de borrarlo y crear uno nuevo con la nueva información. De esta manera, podrá guardarlo tantas veces como quiera con el nombre que prefiera. En la figura 3.29 se pueden apreciar las distintas opciones comentadas.

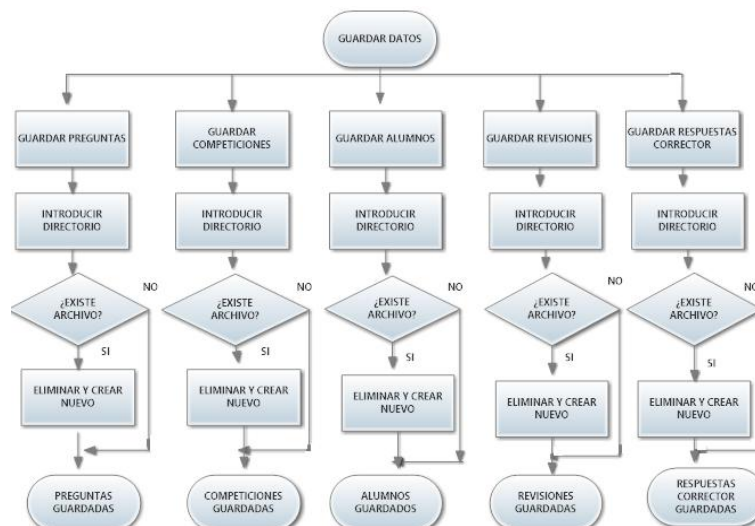


Figura 3.29 Guardar datos.

- El menú del profesor que es administrador contiene las opciones anteriormente descritas y además:

- En gestionar grupos tiene la posibilidad de crear grupos, ya sean vacíos o mediante un CSV (cómo se describió en el apartado de inicio de sesión del administrador) y también cuenta con la posibilidad de borrar un grupo. En la figura 3.30 se representa la parte nueva que es la de borrar grupo, ya que el proceso de crear grupos y editarlos, se vio en la sección de inicio de sesión y en el apartado gestionar grupos.

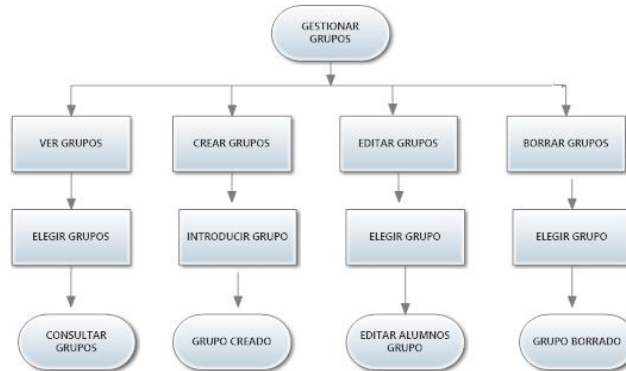


Figura 3.30 Gestionar grupos administrador.

- Gestionar profesores: dentro de esta opción se encuentra:
 - Añadir profesores: el profesor administrador podrá crear más profesores para adjudicarles un grupo o varios.
 - Editar profesores: el profesor administrador podrá editar los datos del profesor (nombre, apellidos, grupo) o eliminar al profesor si lo desea. Este proceso es similar al editar alumnos de gestionar grupos.
 - Ver profesores: el profesor administrador podrá listar los profesores pertenecientes a cada grupo que haya creado.

En la figura 3.31 se presenta todas las opciones posibles.

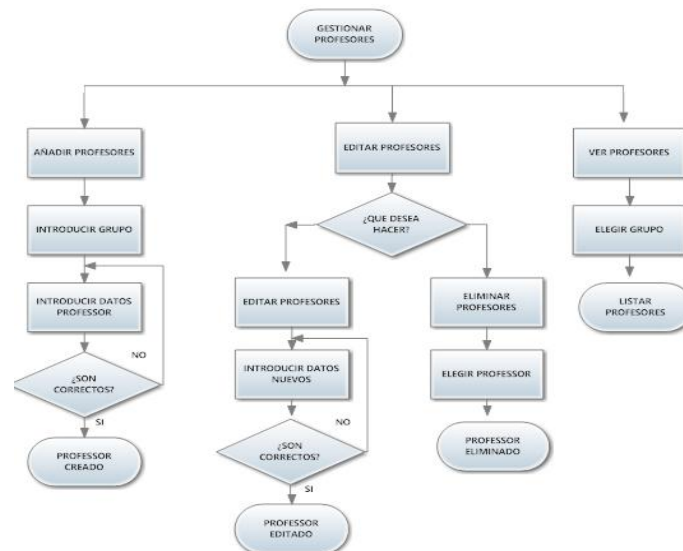


Figura 3.31 Gestionar profesores.

- El menú del alumno contará con las siguientes opciones:

- Mis datos: el alumno podrá consultar sus datos (nombre, apellidos, nia, login y grupo al que pertenecen). En la figura 3.32 se representa dicha opción.



Figura 3.32 Mis datos.

- Competiciones: el alumno podrá acceder a las competiciones que estén abiertas y a las que pueda jugar. La figura 3.33 presenta el flujo del proceso.



Figura 3.33 Competiciones alumno.

- Competiciones cerradas: el alumno podrá consultar las competiciones que ya han acabado. En la figura 3.34 se aprecia dicho proceso.



Figura 3.34 Competiciones cerradas alumno.

- Resultados: el alumno podrá consultar los resultados obtenidos en cada fase que serán:
 - Resultados fase escritor: el alumno podrá consultar el resultado obtenido en esta fase. La nota se obtiene a partir de la nota que el profesor haya puesto a la pregunta creada por el alumno en su fase de corrección de profesor y por la puntuación que los alumnos le han puesto a la pregunta en la fase revisor.
 - Resultados fase revisor: el alumno podrá consultar el resultado obtenido en esta fase. La nota se obtiene a partir de la nota que el profesor haya puesto a la revisión del alumno en su fase de corrección de profesor y por la puntuación que los alumnos le han puesto a la revisión en la fase de corrector.
 - Resultados fase corrector: el alumno podrá consultar el resultado obtenido en esta fase. La nota se obtiene a partir de la nota que el profesor haya puesto a la pregunta cambiada por el alumno (en fase corrector) en su fase de corrección de profesor.
 - Fallos/Aciertos juego inicial: el alumno podrá consultar los fallos y aciertos obtenidos al contestar las preguntas de la fase revisor. Además, podrá listar estas preguntas de forma detallada para que tenga conocimiento de cuáles ha acertado y cuáles fallado.
 - Fallos/Aciertos juego final: el alumno podrá consultar los fallos y aciertos obtenidos al contestar las preguntas de la fase juego. Además, podrá listar estas preguntas de forma detallada para que tenga conocimiento de cuáles ha acertado y cuáles ha fallado.

En la figura 3.35 se representan todos los tipos de resultados que hay.

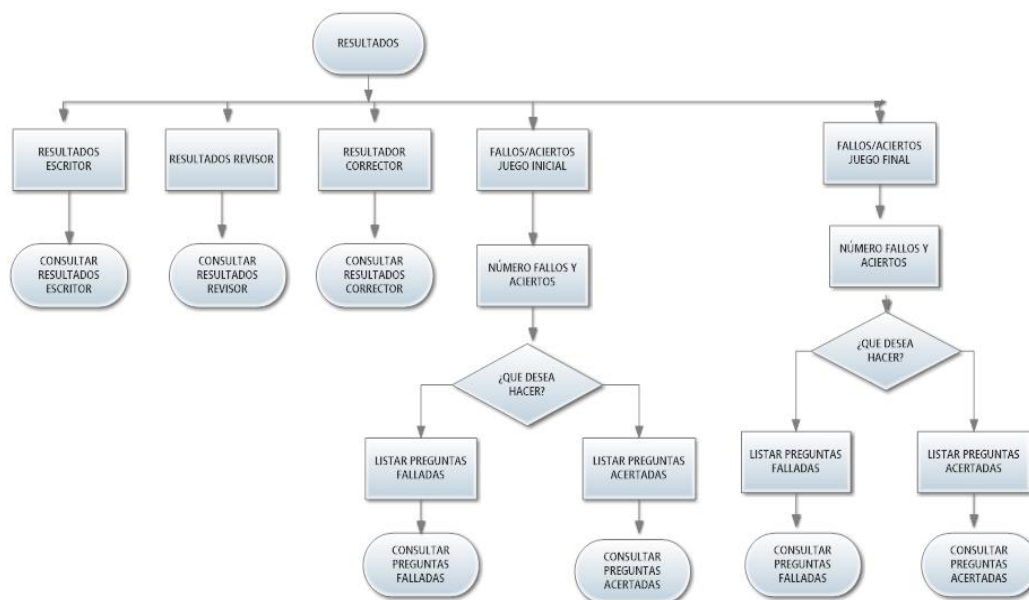


Figura 3.35 Resultados alumno.

- Preguntas corregidas profesor: el alumno podrá consultar las preguntas que ha creado en la fase escritor y si han sido corregidas o no por el profesor en su fase de corrección. Además, se podrán ver de forma

detallada (enunciado, respuestas y opción correcta). En la figura 3.36 se representa dicho proceso.



Figura 3.36 Preguntas corregidas profesor

- Cambiar contraseña: el alumno podrá cambiar de contraseña cada vez que lo desee. La contraseña deberá cumplir una serie de requisitos para aportar más seguridad. Este proceso sigue el mismo flujo que se ha representado en la figura 3.30.
- Guía del juego: el alumno podrá consultar la guía del juego, donde se explica en que consiste el juego y como se realiza la competición. La figura 3.37 hace referencia a esta opción.



Figura 3.37 Guía del juego.

3.6.6 Creación competición

Una vez se hayan registrado los alumnos, se podrá crear una competición. Para ello, se deberá elegir el grupo donde se quiera crear la competición y se introducirán los datos de esta, como las fechas de cada fase teniendo en cuenta que el inicio de una fase es el final de otra y también el número de preguntas y revisiones que tendrán que realizar en la competición. Finalmente, se hará una validación de estos datos en caso de que no se esté seguro o en caso de que se hayan introducido mal fechas y de este modo poder modificarlo antes de validarlo. Cuando todos los datos estén correctos, se procederá a validar la competición. En la figura 3.38 se aprecia el flujo que sigue dicho proceso.

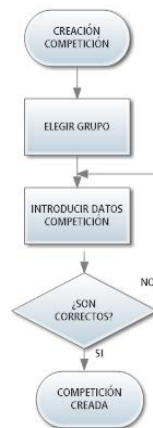


Figura 3.38 Creación competición.

3.6.7 Realizar competición

Cuando la competición esté creada correctamente con las fechas de cada fase y el número de preguntas y revisiones para cada una, se procederá a realizar la competición por parte del alumno.

El alumno deberá acceder a su lista de competiciones y seleccionar la que desee jugar. En este punto nos encontramos con varias situaciones:

- En caso de que aún no sea la fecha de comienzo, no podrá iniciar la competición ya que aún no se encuentra abierta.
- En caso de que la fecha actual se encuentra entre la de inicio y la de siguiente fase, o es la misma que la de inicio, podrá acceder al juego ya que aún tiene tiempo de poder realizar la primera fase.
- En caso de que intente acceder al juego en una fecha que es pasada a la de segunda fase, tampoco podrá acceder ya que se encuentra fuera de fecha.

Si logra acceder a la competición, podrá realizar la fase y pasar a la siguiente. Una vez en la siguiente tendrá que esperar a su inicio y lo mismo se repite para las demás fases del alumno. En la figura 3.39 se puede apreciar el flujo que se realiza al realizar una competición.

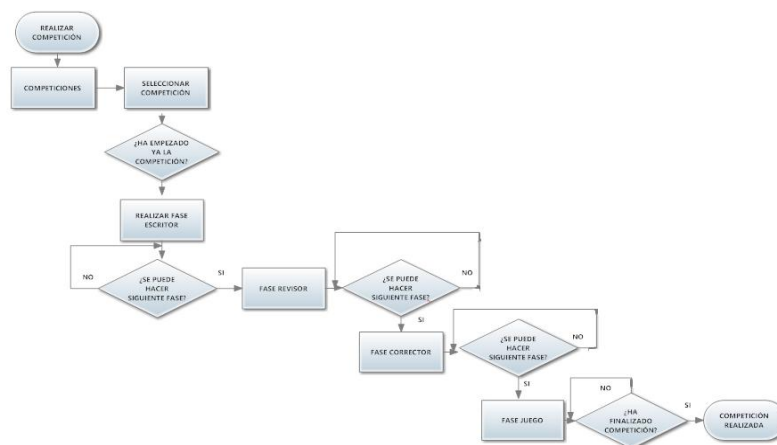


Figura 3.39 Realizar competición alumno.

En el caso del profesor, pasará exactamente lo mismo. No podrá empezar a realizar su fase hasta que no llegue la fecha indicada y si está fuera de fecha, no podrá acceder a la competición. La figura 3.40 representa el caso del profesor.

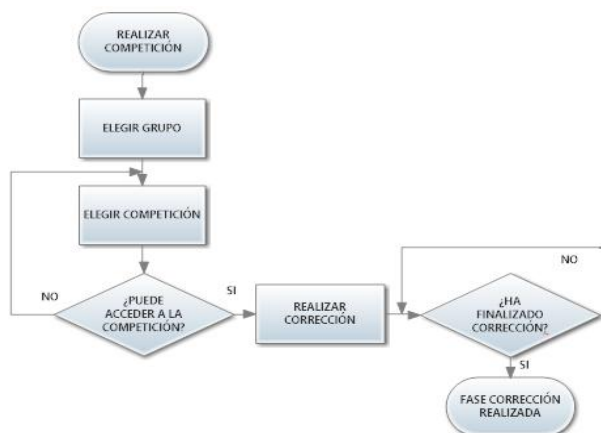


Figura 3.40 Realizar competición profesor.

3.6.8 Finalizar competición

Una vez se hayan completado todas las fases, tanto por parte de los alumnos como por parte del profesor, y se haya acabado el tiempo de acceder a la competición significará que ésta habrá finalizado.

Los alumnos serán capaces de ver los resultados obtenidos (aquellos mencionados anteriormente en la sección del menú del alumno). Como se mencionó, podrán consultar todas sus preguntas, revisiones y todo lo que han ido haciendo en la competición para, de esta manera, tener conocimiento de que debe o no debe mejorar para competiciones futuras. En la figura 3.41 se hace una breve representación de lo que se puede consultar una vez finalizada la competición.

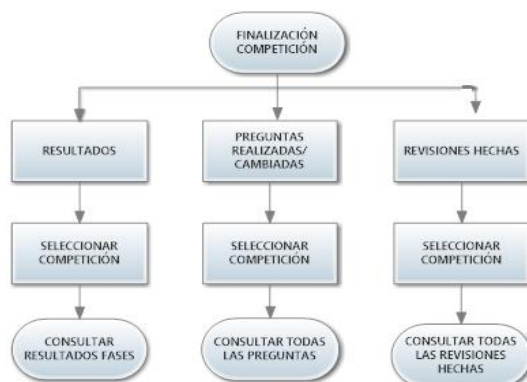


Figura 3.41 Finalización competición alumno.

Los profesores serán capaces de ver los resultados de los alumnos listados por grupo (aquellos mencionados en la sección del menú del profesor). Además, podrá consultar todo lo creado por el alumno que desee y así tener un mejor conocimiento de cómo han ido mejorando o empeorando los alumnos, si han mostrado interés por completar todas las fases, si las preguntas que han creado han sido de un nivel aceptable, si las revisiones han sido creadas desde un punto de vista objetivo, etc. Por otra parte, también

podrá acceder a las preguntas finales y ver los alumnos que han acertado o fallado estas, tal y como se dijo anteriormente y si desea puede exportar la lista de preguntas en un CSV para tenerlas guardadas y que así pueda hacer lo que desee con estas. La figura 3.42 representa las posibilidades que tiene el profesor al finalizar la competición.

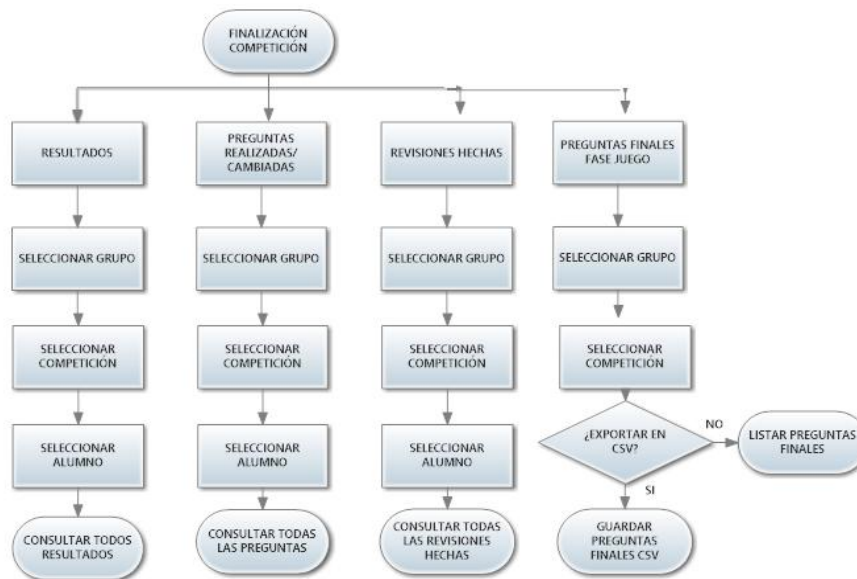


Figura 3.42 Finalización competición profesor.

3.7 RESUMEN DEL CAPÍTULO

En este capítulo se han explicado los pasos que se han llevado a cabo para el análisis, diseño y desarrollo de la aplicación.

En primer lugar, se han explicado los casos de uso que presenta el trabajo para tener un conocimiento de los diferentes escenarios que se van a tener en la aplicación en relación con el usuario que la esté utilizando, ya sea un alumno, profesor o administrador.

A continuación, se han definido los requisitos que tiene que seguir la aplicación para poder cumplir con los objetivos propuestos para la realización del trabajo.

En el siguiente apartado se ha explicado la dinámica del juego, donde se han definido las diferentes fases y se ha hecho un resumen del juego.

En la parte de modelo de datos, se ha explicado los modelos de la aplicación que son fundamentales para saber la estructura lógica que seguirán estos. Además, se ha hecho un desarrollo de las clases que forman dicha aplicación y que son necesarias para su funcionamiento. A esto hay que añadirle que se ha explicado la base de datos, viendo en detalle las tablas que la forman y la información que contendrá cada una de estas tablas.

Por último, se han hecho diagramas de flujo para representar el funcionamiento que seguirá la aplicación detallando las diferentes posibilidades que tienen para interactuar el usuario (alumno, profesor o administrador) con la aplicación.

Capítulo 4

VALIDACIÓN DE REQUISITOS

En esta sección lo que se va a hacer es verificar y revisar el funcionamiento correcto de los requisitos que se mencionaron en el capítulo de diseño e implementación.

Para ello, se hará uso de una serie de capturas de pantallas que se obtienen a través de unas pruebas realizadas de la aplicación cuando está ejecutándose. Esto ayudará a tener una mejor visión de cómo funciona la aplicación cuando es usada por el usuario, ayudando a comprender mejor la funcionalidad que presenta y alejándose de una descripción más técnica.

4.1 ACCESO A LA APLICACIÓN

Esta página es común para todos los usuarios que accedan a la aplicación. Dentro se pueden realizar varias opciones.

4.1.1 Acceder a la aplicación (requisitos cumplidos: 1 y 3)

La primera visión que se tendrá de la aplicación, será la pantalla de inicio de sesión tal y como se muestra en la figura 4.1.

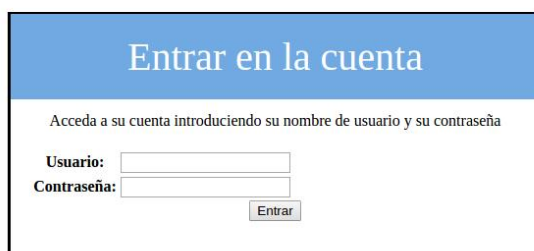
The image shows a login form with a blue header bar containing the text "Entrar en la cuenta". Below the header, there is a light gray box with the text "Acceda a su cuenta introduciendo su nombre de usuario y su contraseña". Inside this box, there are two input fields: "Usuario:" and "Contraseña:". To the right of the "Contraseña:" field is a small "Entrar" button.

Figura 4.1. Acceso a la aplicación.

Se pueden diferenciar 2 formas diferentes de acceder a la aplicación, según el usuario que sea:

- Acceso a un alumno.

Para acceder a la aplicación deberá introducir su NIA como usuario y contraseña. Una vez dentro, aparecerá la página principal de éste con las posibles opciones que puede realizar.

En la figura A.1 del apéndice se puede ver el login, en este caso su NIA, que el alumno debe introducir para acceder.

- Acceso al profesor.

El procedimiento es el mismo que en el acceso al alumno. El profesor (administrador o no) deberá introducir el login, en este caso un alias, y la contraseña. En la figura A.2 se muestra el acceso de un profesor utilizando un alias.

En caso de que se introduzca la contraseña incorrecta o se escriba un login que no existe, la aplicación informará de ello. Las figuras A.3 y A.4 presentan dichas situaciones.

4.1.2 Cambiar de contraseña (requisito cumplido: 2)

El usuario deberá cambiar la contraseña la primera vez que acceda a la aplicación. Si cumple con los requisitos que la contraseña necesita, el cambio se realizará dejando al usuario poder acceder a la página principal. Si no es la primera vez, podrá realizar el cambio cuando desee. La figura A.5 muestra el formulario de cambio de contraseña, y la figura A.6 informa de que la contraseña se ha cambiado correctamente, ambas presentadas en el apéndice.

En caso de que se introduzca una contraseña actual errónea o que la nueva contraseña que se desee introducir no cumpla con las condiciones que se explican, la aplicación devolverá un mensaje de error avisando de la situación. Las figuras A.7 y A.8 del apéndice muestran las situaciones explicadas.

4.2 MENÚ OPCIONES

Una vez que se acceda a la aplicación, la página que se va a mostrar es la del menú de opciones que los usuarios pueden realizar dentro de la aplicación. Según el usuario que acceda a la aplicación se puede dividir en:

- Menú opciones del alumno.
El alumno dispondrá de una página principal diferente del de los profesores donde puede realizar varias opciones. La figura 4.2 muestra las opciones del alumno.

Bienvenido/a 100100100



[Mis datos](#) | [Competiciones](#) | [Competiciones cerradas](#) | [Resultados](#) | [Preguntas corregidas profesor](#) | [Cambiar contraseña](#) | [Guía del juego](#) | [Acerca de](#)

Figura 4.2 Página principal del alumno

- Menú opciones del profesor no administrador.
La página principal del profesor no administrador es diferente del de los alumnos. La figura 4.3 muestra las opciones del profesor no administrador.



Figura 4.3 Página principal del profesor no administrador

- Menú opciones del profesor administrador.
La página principal del profesor administrador, es ligeramente diferente a la del profesor no administrador. La figura 4.4 muestra las opciones del profesor administrador.



Figura 4.4 Página principal del administrador.

4.3 GESTIONAR GRUPOS

El profesor administrador y no administrador podrán acceder a la gestión de grupos. La única diferencia que hay son las opciones que pueden hacer cada uno de ellos.

En la figura A.9 se muestran las opciones que tiene el profesor administrador para la gestión de grupos, y la figura A.10 muestra las opciones del profesor no administrador para la misma gestión, ambas figuras presentadas en el apéndice.

4.3.1 Consultar alumnos de un grupo (requisitos cumplidos: 10, 29)

El profesor (administrador o no) podrá consultar la lista de alumnos que hay en un grupo, seleccionado previamente por él. La figura 4.5 muestra la selección del grupo y la 4.6 muestra la lista de alumnos de ese grupo.

Seleccione un grupo

Grupo: 91 ▼

Figura 4.5 Seleccionar grupo

Lista de alumnos del grupo 91

NIA	Login	Nombre	Apellidos
100100100	100100100	Antonio	Gómez Pérez

Figura 4.6 Listado alumnos

Si el grupo que se ha elegido para consultar está vacío y no presenta alumnos, se informará al usuario a través de un mensaje. La figura A.11 del apéndice muestra dicho mensaje.

4.3.2 Editar grupos

Dentro de esta opción se tienen otras más que se pueden realizar en el grupo. La figura A.12 del apéndice muestra estas opciones.

4.3.2.1 Añadir alumnos a partir de CSV (requisitos cumplidos: 7, 4, 17)

En esta opción, el profesor podrá añadir alumnos a través de un archivo CSV con un formato especial. Para ello, tendrá que escoger el archivo que desee y cargarlo como se ve en la figura 4.7 y luego deberá introducir el grupo al que desea añadir los alumnos, como muestra la figura 4.8.

Añadir alumnos importando
archivo

INTRODUCIR ARCHIVO PARA CARGAR ALUMNOS

Seleccionar archivo

Ningún archivo seleccionado

Importar

Figura 4.7 Importar archivo CSV

¿Son correctos estos datos?

NIA	Login	Nombre	Apellidos
100282983	100282983	Grecia	Vera Villajulca
100285003	100285003	Alexandre	Solves Almeida
123456789	123456789	Pedro	Gutierre Perlin
111222333	111222333	Ivan	Rodriguez Pereda
111111111	111111111	Sandra	García García

Grupo:

91 ▾

Sí

No

Figura 4.8 Listar alumnos del CSV y añadir al grupo

4.3.2.2 Añadir alumnos de forma manual (requisitos cumplidos: 6, 4, 17)

El profesor podrá añadir a un alumno de forma manual. Primero se deberá seleccionar el grupo en el que quiere añadirse el nuevo alumno y luego rellenar los datos del alumno. La figura 4.9 muestra el formulario con los campos necesarios para crear a un nuevo alumno.

Dar de alta a un nuevo alumno

Introducir datos del alumno:

NIA:

Contraseña:

Nombre:

Apellidos:

Figura 4.9 Añadir alumno manualmente

4.3.2.3 Editar alumno de un grupo (requisitos cumplidos: 8, 18)

El profesor podrá editar los datos de los alumnos. Para ello, deberá escoger primero el grupo para listar a los alumnos pertenecientes a él, tal y como se muestra en la figura A.13 del apéndice.

Si se selecciona el botón del NIA, aparecerán las opciones de editar y eliminar, representadas en la figura A.14.

Si se elige editar, se podrán modificar los datos del alumno excepto el NIA, que será único para cada alumno. La figura 4.10 muestra el formulario a rellenar para editar al alumno.

NIA	Nombre	Apellidos	Grupo
100100100	Antonio	Gómez Pérez	91

NIA:

Nombre:

Apellidos:

Grupo:

Figura 4.10 Editar alumno

4.3.2.4 Eliminar alumno del grupo (requisitos cumplidos: 9, 19)

Partiendo de la figura A.14, si se desea eliminar al alumno se selecciona el botón eliminar. Aparecerá otra pantalla en la que se pregunta si se está seguro de la opción, como indica la figura 4.11.

Validar eliminar alumno

¿Seguro que desea eliminar al siguiente alumno?

NIA	Login	Nombre	Apellidos	Grupo
100100100	100100100	Antonio	Gómez Pérez	91

Si

No

Figura 4.11 Validar eliminar alumno

4.3.3 Crear grupo

Esta opción, tal y como se comentó anteriormente sólo puede realizarla el profesor administrador. Se dispone de dos tipos de creación de grupo, mostradas en la figura A.15 del apéndice.

4.3.3.1 Crear grupo vacío (requisito cumplido: 14)

Para crear un grupo vacío simplemente hay que introducir el valor del grupo. La figura 4.12 muestra este formulario.

INTRODUCIR EL GRUPO

Grupo:

Crear grupo

Figura 4.12 Crear grupo vacío

En caso de que el grupo que se introduzca, la aplicación dará un aviso de que ya existe y deberá volver a escribirme el grupo. La figura A.16 muestra el mensaje que enviaría la aplicación.

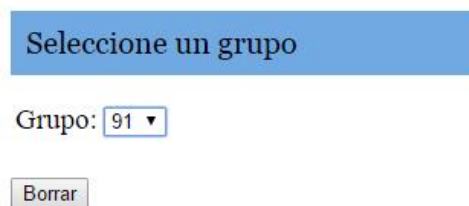
4.3.3.2 Crear grupo importando alumnos (requisito cumplido: 15)

Si se elige la opción de crear grupo importando alumnos se deberá importar el archivo CSV y añadir el grupo tal y como se muestran en las figuras A.17 y A.18 respectivamente y que son similares a las figuras 4.7 y 4.8.

En caso de que el grupo que se agregue ya exista, aparecerá el mismo mensaje mostrado en la figura A.16.

4.3.4 Borrar grupo (requisito cumplido: 16, 9, 33, 45)

El profesor administrador podrá borrar el grupo que desee. Para ello, tendrá que primero elegir el grupo que desee y luego seleccionar borrar. La figura 4.13 muestra la opción de borrar.



Seleccione un grupo

Grupo: 91 ▼

Borrar

Figura 4.13 Borrar grupo

4.4 GESTIONAR PROFESORES

El profesor administrador será el que pueda acceder a la gestión de profesores. En la figura A.19 del apéndice se muestran las opciones que se pueden realizar con los profesores.

4.4.1 Añadir profesor (requisitos cumplidos: 5, 42)

El profesor administrador podrá añadir a un nuevo profesor. Primero deberá seleccionar el grupo en el que quiere crear el nuevo profesor y luego rellenar sus datos. La figura A.20, similar al de creación de un alumno, muestra el formulario con los campos necesarios para crear a un nuevo profesor. En este formulario existe la opción de elegir si el profesor que se está creando será o no administrador.

4.4.2 Editar profesores

En esta opción se podrá tanto editar los datos de los profesores, como eliminarlos del grupo si se desea.

4.4.2.1 Editar profesor del grupo (requisito cumplido: 44, 43)

Se trata de editar los datos del profesor. Para ello, se seleccionará el grupo al que pertenece el profesor que se desea. Con esto se listarán los profesores presentes en ese grupo y se podrán seleccionar. Una vez seleccionados, aparecerán dos opciones que son editar o eliminar, de igual manera que se menciona en el apartado 4.3.2.3. Se podrán editar todos los campos a excepción del login, que será único para cada profesor. La figura A.21 muestra el formulario para editar al profesor, y que es similar a la figura 4.10.

4.4.2.2 Eliminar profesor del grupo (requisito cumplido: 45)

El proceso de eliminar profesor es igual que el explicado en el 4.3.2.4.

4.4.3 Ver profesores (requisitos cumplidos: 11, 43)

El profesor administrador podrá consultar los profesores pertenecientes a un grupo. El proceso de consulta será igual que para la consulta de alumnos del punto 4.3.1. En la

figura A.22 se muestran los datos de un profesor y en la figura A.23 se muestra el caso en el que no existan profesores en el grupo seleccionado.

4.5 GESTIONAR COMPETICIONES

El profesor (administrador o no) podrá gestionar las competencias. La figura A.24 del apéndice, muestra las opciones que se pueden realizar para la gestión de las competencias.

4.5.1 Listar todas las competencias (requisito cumplido: 34)

En este caso el profesor puede consultar todas las competencias del grupo que seleccione previamente. En esta lista de competencias están aquellas que se encuentran abiertas, y aquellas que se encuentran cerradas. Las figuras A.25 y A.26, representa la selección del grupo y la lista de competencias de ese grupo.

4.5.2 Listar competencias abiertas (requisito cumplido: 35, 62, 63)

En este caso, el usuario podrá listar aquellas competencias que se encuentren abiertas. El proceso es el mismo explicado en el apartado 4.5.1, lo único que cambia es el filtro que se utiliza para devolver la información.

4.5.3 Listar competencias cerradas (requisito cumplido: 36, 61)

En este caso el usuario (alumno y profesor) solamente podrá listar aquellas competencias que se encuentren cerradas. El proceso es el mismo explicado en el apartado 4.5.1, lo único que cambia es el filtro que se utiliza para devolver la información. En caso de que no haya competencias cerradas, la figura A.27 muestra el mensaje que la aplicación da al usuario.

4.5.4 Corregir preguntas finales alumnos (requisitos cumplidos: 41, 46, 31, 57, 58, 54, 55, 60)

Se trata de la fase de corrección del profesor. Como se explicó anteriormente, el profesor (administrador y no administrador) deberá validar las preguntas que pasen a la fase final.

Para poder acceder a la competición, deberá seleccionar el grupo del que quiere consultar la competición y deberá asegurarse de que esta se encuentre abierta. La figura A.28 muestra la pantalla de acceso a la fase de corrección del profesor.

Una vez dentro, se listarán los alumnos que hayan hecho la fase anterior y que hayan creado y/o actualizado preguntas. Un ejemplo se muestra en la figura A.29. Como se ha podido apreciar, se puede seleccionar el NIA del alumno. Si se selecciona, aparecerá la lista de preguntas que ha creado el alumno, con información de si se ha hecho corrección y si se ha modificado en fase corrector. La figura A.30 muestra la tabla

detallada. En caso de que no haya alumnos que hayan hecho la fase corrector, la aplicación informara de esto, tal y como se muestra en la figura A.31.

Como se puede apreciar, en la columna correspondiente al profesor, aparece el enunciado de la pregunta de fase corrector en caso de que se haya sido cambiada, o el enunciado de la fase escritor en caso de que no. Esto es debido a que el profesor aún no ha corregido la pregunta, por lo que dicha pregunta sigue igual.

Si lo desea, el profesor puede cambiar también la pregunta al momento de validarla. Para ello, deberá puntuar la pregunta o preguntas y las revisiones asociadas a esta.

Se van a tener dos tipos distintos de corrección de profesor. Por un lado, si la pregunta inicial no ha sido cambiada en la fase de corrector se puntuará ésta junto con sus revisiones correspondientes. La figura 4.14 presenta la página de la fase de corrección del profesor.

Pregunta con las revisiones realizadas

Enunciado pregunta:
Enunciado pregunta 2 alumno 2 para fase escritor

Opción 1:
respuesta 1 alumno2

Opción 2:
respuesta 2 alumno2

Opción 3:
respuesta 3 alumno2

Opción 4:
respuesta 4 alumno2

Opción correcta:
respuesta 4 alumno2

Puntuación a la pregunta

¿Desea corregir la pregunta?

Nota: Si pulsa no, las notas se guardaran automaticamente.

Puntuación de la pregunta: 7.0

Comentario: comentario alumno1 pregunta 1

Posible mejora: posible mejora alumno1 pregunta 1

Puntuación a la revision

COMENTARIO CORRECTOR:
comentario alumno 2 a la revision

Figura 4.14 Fase corrección profesor

Por otro lado, si la pregunta inicial si ha sido cambiada en la fase corrector, se deberán puntuar las dos preguntas, es decir, la pregunta creada en la fase escritor y la pregunta creada en la fase corrector. La página para esta fase es similar al de la figura 4.14, solo que se le añade aquella pregunta inicial, como se muestra en las figuras A.32 y A.33.

Por lo demás, el funcionamiento es igual para ambos casos. En caso de que desee cambiar la pregunta, aparecerá el formulario de creación de preguntas igual que el de la fase escritor. Antes de realizar dicho cambio, deberá puntuar las preguntas y las revisiones, ya que si no lo hace no podrá cambiarla. Una vez cambiada y validada la pregunta, se volverá a la lista de preguntas, donde aparece el nuevo enunciado de la pregunta cambiada por el profesor, tal y como se muestra en la figura A.34 del apéndice. Si decide no cambiarla y validarla, no aparecerán cambios en la lista de

preguntas. Para ambos casos, aparecerá un mensaje informando que las puntuaciones se han guardado correctamente y volverá a la lista de preguntas. La figura A.35 muestra el mensaje de confirmación.

De esta forma pasaran a la fase juego aquellas preguntas que estén validadas por el profesor. Si se ha cambiado, pasará a la fase juego la pregunta del profesor. Si no se ha cambiado, pasara la pregunta del alumno validada.

Cuando el profesor termine de realizar su fase, tendrá la opción de validar todo para dar por finalizada su fase. Si selecciona dicho botón y quedan alumnos que no ha corregido, aparecerá un mensaje informando del alumno o alumnos que están aún sin validar. Esto se muestra en la figura A.36.

Por el contrario, si todos los alumnos han sido validados, pasará al final de su fase. La figura A.37 muestra el fin de la fase de corrección de profesor.

Una vez que el profesor ha terminado y la fecha actual coincide con la de inicio fase juego, se podrá acceder a la competición para finalizarla.

4.5.5 Consultar preguntas corregidas por el profesor (requisitos cumplidos: 66, 67, 68)

En este apartado el alumno podrá consultar las preguntas que el profesor le ha corregido, pudiendo consultar de igual manera las suyas iniciales y las que ha cambiado en la fase corrector.

Para acceder a estas, deberá seleccionar la competición de la cual quiere obtener las preguntas. Una vez seleccione la competición, le aparecerá las preguntas que él mismo creó inicialmente, indicando si se han corregido en la fase corrector y si las ha corregido o no el profesor. De esta manera puede consultar tanto las preguntas creadas al principio, como las cambiadas por él mismo o por el profesor. La figura 4.15 muestra cómo aparecería la lista de preguntas al alumno y sus correcciones.

Pregunta inicial	Corregida	Pregunta final	Corregida Profesor	Pregunta final profesor
Enunciado pregunta 1 alumno2 para fase escritor	SI	pregunta 1 cambiada alumno2 fase corrector	NO	pregunta 1 cambiada alumno2 fase corrector
Enunciado pregunta 2 alumno 2 para fase escritor	NO	Enunciado pregunta 2 alumno 2 para fase escritor	SI	pregunta 2 alumno 2 cambiada profesor

Figura 4.15 Lista de preguntas y correcciones

Como se puede comprobar, el alumno podrá seleccionar cualquier pregunta en caso de que esté corregida para que pueda ver en detalle las preguntas (figura A.51) y poder comparar o sacar conclusiones de su respuesta inicial con respecto a la final.

4.5.6 Jugar competición (requisito cumplido: 41)

Es el requisito principal de la aplicación. El alumno debe ser capaz de jugar la competición realizando sus 4 fases en las fechas correspondientes. Para ello, primero deberá seleccionar la competición que desee jugar, como se muestra en la figura A.38.

4.5.6.1 Fase escritor (requisito cumplido: 64)

El alumno empezará realizando la fase escritor. Una vez elegida la competición, se comprobará la fecha en la que se encuentra el usuario y la fecha de todas las fases. Si el usuario se ha quedado en alguna fase, se comparará la fecha actual con la fecha en la que empieza la fase que le corresponde realizar. Si intenta acceder antes, se le mostrará un mensaje diciendo que aún no puede acceder, como se muestra en la figura A.39.

Por el contrario, si intenta acceder a la competición y se encuentra dentro de las fechas correctas podrá entrar en la competición y empezarla o seguirla desde la fase en la que dejó el juego.

En este caso, se empezará por la primera fase. La figura A.40 muestra el caso en el que el alumno ha entrado para realizar la fase escritor y así poder comprobar que está en la fecha correcta.

Cuando seleccione el botón de competir, entrará al juego y se le mostrará un formulario para la creación de preguntas donde hay que poner el enunciado, escribir las cuatro posibles respuestas y seleccionar la respuesta que será la correcta. Este formulario se muestra en la figura 4.16.

The screenshot shows a web interface for the 'Fase ESCRITOR' (Writer Phase). At the top, there is a blue header bar with the text 'Fase ESCRITOR'. Below this, a message states: 'Usted deberá escribir 2 preguntas con múltiples respuestas' (You will have to write 2 questions with multiple answers). The form consists of several input fields: a large text area for 'Enunciado pregunta:' (Question statement), followed by four smaller text areas for 'Opción 1:', 'Opción 2:', 'Opción 3:', and 'Opción 4:'. At the bottom, there is a dropdown menu labeled 'Opción respuesta:' with 'Opcion 1' selected. Each input field has a small icon in the bottom right corner, likely for clearing the field.

Figura 4.16 Formulario de las preguntas

Se puede observar que en la breve descripción inicial aparece el número de preguntas que se deben crear. En este caso son dos, por lo que habría que rellenar los dos formularios que aparezcan con las preguntas y respuestas. Una vez creada las preguntas, se presenta la opción de validarlas. Esto sirve como una confirmación de las preguntas creadas, por si en caso de detectar algún fallo en estas, poder cambiarlas. La figura A.41 muestra la opción de validación.

La acción de validar sirve para que el usuario pueda asegurarse de que lo que ha hecho está bien. En caso de no validarlo, volverá al formulario. Si lo valida, pasará a realizar la siguiente fase, la fase de revisor.

4.5.6.2 Fase revisor (requisito cumplido: 54)

Se trata de la segunda fase de la competición. Si el alumno accede a esta, se le mostrará la figura A.42 que le informa de la fecha de comienzo de dicha fase.

En la fase revisor aparecerá una lista de preguntas para ser respondidas. Las preguntas de la lista son aleatorias y el número de preguntas que aparecen viene determinada en el momento en que se creó la competición. La forma aleatoria está implementada de tal forma que se asegure que al alumno que hace la revisión, no le toque revisar una pregunta que él mismo ha creado. Además, está realizado de tal forma que primero se seleccionarán aquellas que no tienen ninguna revisión, escogiendo de entre estas aleatoriamente las preguntas. La figura 4.17 muestra una pregunta con sus posibles respuestas de las que se tendrá que escoger una opción junto con la parte en la que se tiene que puntuar dicha pregunta, comentarla y evaluar su dificultad, si está bien explicada o mal formada y escribir una posible mejora a esta pregunta.

Fase REVISOR

Deberá jugar 2 preguntas y revisarlas, puntuandolas y comentandolas.

Enunciado pregunta:
¿Enunciado de pregunta para el juego inicial?

Opción 1:
Aquí se escribe la opción 1

Opción 2:
Aquí se escribe la opción 2

Opción 3:
Aquí se escribe la opción 3

Opción 4:
Aquí se escribe la opción 4

Opción correcta: Opcion 1 ▼

Puntuación de la pregunta

Comentario

Posible Mejora

Figura 4.17 Fase revisor

Cuando haya escritor los comentarios y puntuado las preguntas, accederá a la siguiente fase, la fase corrector.

4.5.6.3 Fase corrector (requisitos cumplidos: 65, 60)

Se trata de la tercera fase de la competición. Si el alumno accede a esta, se le mostrará la figura A.43 informando de la fecha en la que esta empieza.

Si se accede a la fase corrector, se listarán las preguntas creadas al inicio en la fase escritor junto con la puntuación, el comentario a la pregunta y la posible mejora de cada revisión que tenga dicha pregunta. En esta fase lo que debe hacer el alumno es corregir

dichas revisiones. Tiene que puntuar la revisión obtenida y comentar dicha revisión. La figura 4.18 muestra la pregunta y la revisión asociada junto con la puntuación que se le dará a la revisión y el comentario hecho.

Fase CORRECTOR

Deberá corregir sus preguntas iniciales y puntuar a sus revisores

Enunciado pregunta:
Enunciado pregunta 2 alumno 2 para fase escritor

Opción 1:
respuesta 1 alumno2

Opción 2:
respuesta 2 alumno2

Opción 3:
respuesta 3 alumno2

Opción 4:
respuesta 4 alumno2

Puntuación de la pregunta: 7.0

Comentario a la pregunta: comentario alumno1 pregunta 1

Posible mejora: posible mejora alumno1 pregunta 1

Puntuación de la revision

Comentario a la revision

Añade un comentario de la revisión

Cambiar pregunta

Validar

Nota: Pulse cambiar pregunta si desea corregirla

Figura 4.18 Fase corrector

El alumno tendrá la opción de cambiar la pregunta, es decir, volver a escribirla si lo desea. En el caso de hacerlo, aparecerá el mismo formulario que en la fase escritor. Una vez que se cambie la pregunta y se valide, volverá a la página de la fase corrector. Deberá puntuar la revisión y añadir un comentario para poder validar dicha corrección. Una vez validada, esta se quitará de la lista de preguntas, quedando aquellas preguntas y revisiones que aún no se han corregido. Si no se valida, se volverá a la página de fase corrector. Las figuras A.44 y A.45 muestran el proceso del cambio de pregunta.

Cuando se haya dado la puntuación y se haya escrito el comentario, se podrá validar la corrección. Si no ha habido ningún problema en la validación, aparecerá un mensaje como en la figura A.46 del apéndice.

Una vez validadas todas las correcciones, aparecerá un botón para poder pasar a la fase juego tal y como aparece en la figura 4.19

Fase CORRECTOR

Deberá corregir sus preguntas iniciales y puntuar a sus revisores

Ir a fase juego

Figura 4.19 Ir a fase juego

Cuando se seleccione dicho botón, se pasará a la fase de corrección del profesor que ya se ha explicado en el apartado 4.5.4. Este proceso es oculto y transparente para los alumnos, por lo que a ellos les aparecerá el acceso a la fase juego directamente.

4.5.6.4 Fase juego

Se trata de la última fase de la competición. Si el alumno accede a esta, se le mostrará la figura A.47 del apéndice.

En la fase juego se listarán las preguntas a contestar, de forma aleatoria, como se muestra en la figura 4.20. Una vez que se contesten todas las preguntas, se pasará al final de la fase y por tanto al final de la competición. La figura A.48 muestra el mensaje de final de competición con su respectiva fecha.

JUEGO

Deberá contestar a 2 preguntas

Enunciado pregunta:
pregunta 2 alumno 2 cambiada profesor

Opción 1:
respuesta 1 profesor alumno2

Opción 2:
respuesta 2 profesor alumno2

Opción 3:
respuesta 3 profesor alumno2

Opción 4:
respuesta 4 profesor alumno2

Opción correcta: Opcion 1 ▼

Enunciado pregunta:
pregunta 1 cambiada alumno2 fase corrector

Opción 1:
respuesta 1 alumno2

Opción 2:
respuesta 2 alumno2

Opción 3:
respuesta 3 alumno2

Opción 4:
respuesta 4 alumno2

Opción correcta: Opcion 1 ▼

Jugar

Figura 4.20 Fase juego

4.5.7 Seguimiento de alumnos

En esta parte, el profesor podrá hacer un seguimiento de cada alumno para saber las fases realizadas, las preguntas creadas, revisiones y correcciones hechas y los comentarios realizados. De esta forma el profesor evaluar el interés y la forma en la que el alumno ha realizado cada fase. El seguimiento de los alumnos comprende diferentes opciones, mostradas en la figura A.49.

4.5.7.1 Alumnos que han creado preguntas (requisitos cumplidos: 25, 20)

En esta sección, el profesor podrá listar a los alumnos que han creado preguntas en la fase escritor y el número de preguntas que han creado. En la figura 4.21 se muestran los alumnos con su número de preguntas hechas.

NIA	Nombre	Apellidos	Preguntas hechas
100285003	Alexandre	Solves Almeida	2
123456789	Pedro	Gutierre Perlin	2
111222333	Ivan	Rodriguez Pereda	2

Figura 4.21 Lista alumnos que han hecho preguntas

Si se selecciona el número de preguntas, se podrán listar las preguntas que ha hecho cada alumno y saber si las ha cambiado en la fase de corrector. Esta parte sería similar a la mostrada en la figura A.30 quitando la columna del profesor. La figura A.50 muestra el enunciado de dichas preguntas y si se han corregido o no y su enunciado después de la corrección.

En el caso de la figura A.50, como se puede comprobar el alumno no ha corregido ninguna pregunta inicial. Las preguntas iniciales pueden seleccionarse para obtener su información detallada. La figura A.51 muestra la información detallada de la pregunta.

4.5.7.2 Alumnos que han hecho revisiones (requisitos cumplidos: 26, 21, 56)

En esta sección se presentan dos posibles opciones de consulta, mostradas en la figura A.52.

4.5.7.2.1 Alumnos que han hecho revisiones obligatorias

En este apartado se podrán consultar los alumnos que han hecho revisiones obligatorias, que son aquellas que la competición determina. Se deberá seleccionar el grupo y la competición que se quiera consultar. Hecho esto, aparecerá la lista de alumnos que han hecho dichas revisiones y el número de revisiones, como se puede ver en la figura 4.22.

NIA	Nombre	Apellidos	Revisiones hechas
100285003	Alexandre	Solves Almeida	2
123456789	Pedro	Gutierre Perlin	2

Figura 4.22 Alumnos que han hecho revisiones

Si se selecciona el número de revisiones, se listaran aquellas preguntas que el alumno ha revisado. De esta manera el profesor podrá valorar el nivel de corrección que tiene el alumno. La figura A.53 presenta la lista de preguntas de las cuales se han hecho las revisiones.

Si se selecciona una de las preguntas, se obtiene la pregunta detallada junto con la revisión que el alumno ha hecho a dicha pregunta. La figura A.54 muestra la pregunta y la revisión asociada a esta, similar a la figura 4.14.

En caso de que no haya alumnos que han hecho revisiones obligatorias, la aplicación informará de esto a través de un mensaje que se muestra en la figura A.55.

4.5.7.2.2 Alumnos que han hecho revisiones opcionales

Se seguirá el mismo proceso descrito en el apartado 4.5.7.2.1. La única diferencia es el mensaje que la aplicación da cuando no hay alumnos que han hecho revisiones opcionales, que se muestra en la figura A.56.

4.5.7.3 Alumnos que han hecho correcciones (requisitos cumplidos: 27, 22)

En esta sección simplemente se podrá ver el número de correcciones que ha hecho cada alumno, ya que las preguntas corregidas aparecen en la sección de alumnos que han hecho preguntas. La figura 4.23 muestra los alumnos que han hecho correcciones.

NIA	Nombre	Apellidos	Correcciones hechas
100285003	Alexandre	Solves Almeida	2
123456789	Pedro	Gutierre Perlin	2

Figura 4.23 Alumnos que han hecho correcciones

En caso de que no haya alumnos que han realizado la fase corrector, se mostrará un mensaje informando de esta situación. Este mensaje se muestra en la figura A.57.

4.5.7.4 Alumnos que han jugado (requisitos: 28, 23)

En esta sección podrá consultar los alumnos que han llegado a la fase de juego y el número de preguntas que han jugado. La figura 4.24 muestra un ejemplo de los alumnos y sus preguntas.

Nia	Nombre	Apellidos	Preguntas jugadas
123456789	Pedro	Gutierre Perlin	2

Figura 4.24 Alumnos que han hecho fase juego

Por el contrario, en caso de que no haya alumnos que han hecho la fase juego, aparecerá un mensaje informando de la situación, tal y como muestra la figura A.58

4.5.7.5 Fase que ha realizado cada alumno (47,48, 49, 50, 51, 66, 67, 68, 56, 54, 55)

Con esta opción se puede consultar la fase en la que se encuentra cada alumno. Si se selecciona esta opción, aparecerá la lista de alumnos de la competición que haya seleccionado y podrá ver la fase de cada uno de ellos. Esto se muestra en la figura 4.25.

Nia	Nombre	Apellidos	FASE ESCRITOR	FASE REVISOR	FASE CORRECTOR	FASE JUEGO
123456789	Pedro	Gutierre Perlin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
100285003	Alexandre	Solves Almeida	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
111222333	Ivan	Rodriguez Pereda	<input checked="" type="checkbox"/>			

Figura 4.25 Fase realizada por cada alumno

Además, podrá acceder a lo que el alumno ha realizado en cada fase. Por ejemplo, si seleccionamos la fase escritor del primer alumno, aparecerán las preguntas que ha creado en la fase escritor.

Si se prueba seleccionando la fase corrector del mismo alumno, aparecerán las preguntas que haya cambiado en la fase corrector. Las preguntas que aparecen, podrán ser seleccionadas para verlas en detalle, tal y como se mostró en la figura A.51.

Si se selecciona la fase revisor, el resultado será el mismo obtenido que en la figura 4.38 pero para un alumno, por lo que el proceso para consultar la revisión sería el mismo que se ha explicado en las opciones de revisión.

Si se selecciona la fase juego, se listará el número de aciertos y fallos que ha tenido en la fase juego, como muestra la figura 4.26.

Fecha inicio JUEGO	Fecha fin JUEGO	Aciertos	Fallos
2015-09-17	2015-09-17	<input type="text" value="1"/>	<input type="text" value="1"/>

Figura 4.26 Aciertos y fallos juego

Si se selecciona cualquier número, se podrá consultar la pregunta detallada (como en la figura A.51) o la lista de preguntas que se han acertado o fallado

4.5.7.6 Resultados

Según que usuario sea el que consulte los resultados, estos se dividen en:

4.5.7.6.1 Resultados de los alumnos por parte del profesor (69, 70, 71, 72, 73, 24)

En esta sección los profesores podrán consultar los resultados que ha obtenido cada alumno en cada fase. Deberán consultar las puntuaciones obtenidas en las fases y las preguntas acertadas o falladas que han tenido. La figura 4.27 muestra los resultados de los alumnos.

Nia	Nombre	Apellidos	Resultado fase Escritor	Resultado fase Revisor	Aciertos juego INICIAL	Fallos juego INICIAL	Aciertos juego FINAL	Fallos juego FINAL
100285003	Alexandre	Solves Almeida	7.6	7.28	<input type="text" value="1"/>	<input type="text" value="1"/>	0	0
123456789	Pedro	Gutierre Perlin	7.54	7.72	0	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>

Figura 4.27 Resultado alumnos

Se puede comprobar que el primer alumno no tiene ni aciertos ni fallos finales, por lo que esto indica que no ha jugado la fase juego.

Se tendrá la posibilidad de seleccionar el número de fallos o aciertos del cual le aparecerá la lista de preguntas o pregunta que ha fallado o acertado. Un ejemplo al seleccionar las preguntas falladas se muestra en la figura A.59.

Si se selecciona cualquiera de las preguntas, podrá consultarse de forma detallada como se mostró en la figura A.51.

4.5.7.6.2 Resultados del propio alumno

Los alumnos podrán consultar sus propios resultados. En esta sección se van a dividir los resultados en diferentes opciones, mostradas en la figura A.60.

4.5.7.6.2.1 Resultados como escritor (requisito cumplido: 69)

El alumno podrá consultar la nota que ha obtenido de las puntuaciones que le han dado los alumnos y la que le ha dado el profesor en la fase de corrección de este. Esta puntuación sigue la siguiente fórmula:

$$\text{Puntuación} = (\text{Media puntuación alumnos}) * 0.4 + (\text{Media puntuación profesor}) * 0.6$$

Las puntuaciones de los alumnos cuentan un 40%, mientras que la del profesor cuenta un 60%. En la figura 4.28 se muestra un ejemplo de resultado como escritor.

Fecha inicio ESCRITOR	Fecha fin ESCRITOR	Puntuación total	Media notas profesor	Media notas alumnos
2015-09-17	2015-09-17	7.6	7.33	8.0

Figuras 4.28 Resultado fase escritor

4.5.7.6.2.2 Resultados como revisor (requisito cumplido: 70)

El alumno podrá ver la nota que los compañeros le han dado en base a las revisiones que ha hecho y la nota que el profesor le ha dado cuando ha realizado su fase de corrección al puntuar las revisiones de las preguntas. Al igual que los resultados como escritor, la puntuación obtenida está en la misma forma (40% alumno, 60% profesor). La forma de representar es la misma que la figura 4.28, pero en este caso cambian las fechas de escritor por las de revisor, quedando lo que se muestra en la figura A.61.

4.5.7.6.2.3 Resultados como corrector (requisito cumplido: 71)

En esta sección, la puntuación cambia con respecto a los otros resultados. En este caso, la nota se va a basar solamente en la del profesor. Cuando el profesor realice su fase de corrección, se listarán todas las preguntas del alumno que seleccione. Si el alumno ha cambiado la pregunta inicial de su fase escritor en la fase corrector, aparecerán las dos preguntas para que el profesor pueda puntuar las dos, la que pertenece a la fase escritor y la que pertenece a la fase corrector. Este proceso se ha explicado en el apartado 4.5.4.

Cuando el profesor puntúe las preguntas, la puntuación que el profesor otorgue a la pregunta cambiada, corresponderá con la puntuación como corrector del alumno. Por el

contrario si la pregunta no ha sido cambiada en la fase de corrector, aparecerá únicamente la pregunta inicial de la fase escritor y el profesor no podrá puntuarle como corrector, por lo que no tendrá resultado. Las figuras A.62 y A.63 muestran el caso de que haya tenido puntuación como corrector y de que no tenga puntuación como corrector respectivamente.

4.5.7.6.2.4 Fallos y aciertos juego inicial (requisitos cumplidos: 72, 50)

Esta sección hace referencia a las preguntas que se respondieron en la fase de revisor. Se podrá acceder al número de cada uno y además listar y ver en detalle las preguntas falladas y acertadas. La lista de los alumnos con sus respectivos fallos y aciertos es la misma que se muestra en la figura 4.26. Y el caso en el que se listen las preguntas, sería igual que la figura A.59 pudiendo verlas en detalle como en la figura A.51.

En caso de que no estén disponibles los resultados, se informará mediante un mensaje igual que en la A.64.

4.5.7.6.2.5 Fallos y aciertos del juego (requisitos cumplidos: 51, 73)

En esta sección, los resultados corresponden con los de la última fase de la competición, los de la fase juego. De igual manera que los fallos y aciertos del juego inicial, el alumno podrá listar las preguntas y consultarlas, para poder tener conocimiento de estas. En caso de que no se tengan resultados, aparecerá el mensaje de la figura A.65.

4.5.8 Estadísticas competición

En esta sección el profesor podrá consultar las preguntas finales que han aparecido en la fase juego y los alumnos que han contestado a estas. Esto servirá para poder hacer una estadística de los alumnos que han fallado más o menos preguntas y así saber que pregunta ha resultado más sencilla o más complicada. Existen dos opciones para poder consultar las estadísticas y se muestran en la figura A.66.

4.5.8.1 Estadística detallada competición (requisitos cumplidos: 49, 39, 37)

Esta opción permite, como se ha comentado anteriormente, consultar las preguntas finales y también los alumnos que han contestado a estas. Para ello, se tendrá que seleccionar la competición que se desea consultar. Una vez se accede a dicha competición, se listarán las preguntas de la fase juego y el número de personas que han acertado y el número de personas que han fallado, tal como se muestra en la figura 4.29.

Enunciado pregunta	Numero personas acertadas	Numero personas falladas
¿Enunciado de pregunta para el juego inicial?	0	0
pregunta 1 cambiada alumno2 fase corrector	0	1
pregunta 2 alumno 2 cambiada profesor	1	0
pregunta 2 alumno1 cambiada profesor	0	0

Figura 4.29 Lista preguntas fase juego

Como se puede apreciar, se puede seleccionar el número que aparece en cada lista, para poder listar los alumnos que han fallado o acertado. La figura A.67 muestra la lista de alumnos (en este ejemplo sólo hay 1) con sus datos.

4.5.8.2 Estadística para exportar a CSV (requisitos cumplidos: 49, 39, 37, 38)

Esta opción solamente permite ver las preguntas finales que han pasado a la fase juego, aunque en este caso se pueden descargar dichas preguntas en un archivo CSV.

El proceso sería igual que el anterior, seleccionando la competición que se desea para listar las preguntas. La figura 4.30 muestra las preguntas y la opción de exportarlas.

Enunciado pregunta	Numero personas acertadas	Numero personas falladas
¿Enunciado de pregunta para el juego inicial?	0	0
pregunta 1 cambiada alumno2 fase corrector	0	1
pregunta 2 alumno 2 cambiada profesor	1	0
pregunta 2 alumno1 cambiada profesor	0	0

[Exportar a csv](#)

Figura 4.30 Lista preguntas exportar

Si se selecciona el botón de exportar alumnos, el servidor creará un fichero CSV que se descargará. La figura A.68 muestra el archivo descargado.

4.5.9 Crear competición (requisito cumplido: 32)

En este caso, tanto el profesor administrador como el no administrador podrán crear las competiciones. La figura 4.31 presenta el formulario que se debe rellenar para poder crear una competición.

Crear una nueva competición

Fase ESCRITOR

Fecha inicio:

Preguntas a escribir:

Fase REVISOR

Fecha inicio:

Preguntas a revisar:

Fase CORRECTOR

Fecha inicio:

Preguntas a corregir de la fase ESCRITOR:

Fase CORRECCION PROFESOR

Fecha inicio:

JUEGO

Fecha inicio:

Preguntas a jugar:

Fecha fin:

Figura 4.31 Crear competición

Para crear una competición se deberán detallar las fechas en la que se quiere realizar cada fase y el número de preguntas o revisiones para cada una de ellas. Las fechas tienen que elegirse de tal forma que sea una posterior a otra, es decir, si se intenta poner una fecha que sea anterior a la que se ha escrito primero, la competición no se podrá crear ya que una fase posterior no puede tener una fecha anterior a otra fase que sea anterior.

En caso de que se detecte este problema, la aplicación informaría a través de un mensaje representado en la figura A.69.

En caso de que las fechas sean correctas, la competición se creará sin ningún problema.

4.5.10 Borrar competición (requisito cumplido: 33)

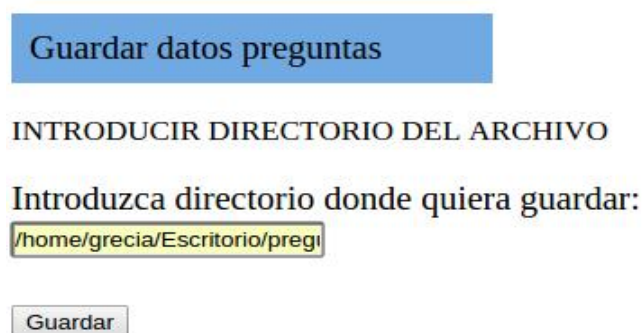
El profesor podrá borrar las competiciones que desee. El proceso es el mismo, se escogerá el grupo y se listarán todas las competiciones de ese grupo. En cada competición aparecerá un botón que permitirá borrar la competición, como se muestra en la figura A.70. Una vez que la competición haya sido borrar, se volverá a la página de inicio.

4.6 GUARDAR DATOS

En esta sección, el profesor será capaz de guardar los datos de la aplicación. De esta manera podrá hacer un backup de todo lo hecho. Para ello tendrá varias opciones de guardado, representados en la figura A.71.

4.6.1 Guardar preguntas (requisito cumplido: 52)

Para guardar los datos de las preguntas de la aplicación, se deberá introducir la ruta donde se desea crear el archivo donde se copiarán las preguntas y se le otorgará el nombre que se desee en formato CSV. La figura 4.32 muestra la forma en la que se debe introducir el directorio del archivo CSV.



Guardar datos preguntas

INTRODUCIR DIRECTORIO DEL ARCHIVO

Introduzca directorio donde quiera guardar:

/home/grecia/Escritorio/pregi

Guardar

Figura 4.32 Directorio donde se guardará archivo

Una vez que el archivo se haya guardado correctamente, la aplicación informará de esto con la figura A.72.

4.6.2 Guardar revisiones (requisito cumplido: 59)

Es el mismo proceso que el descrito en el apartado 4.6.1 con la diferencia de que son revisiones.

4.6.3 Guardar correcciones

Sigue el mismo proceso que en el apartado 4.6.1 con la diferencia de que son correcciones.

4.6.4 Guardar alumnos (requisito cumplido: 12)

Ídem que el apartado 4.6.1 pero con los datos de los alumnos.

4.6.5 Guardar profesores

Ídem que el apartado 4.6.1 pero con los datos de los profesores.

4.6.6 Guardar competiciones (requisito cumplido: 40)

Ídem que el apartado 4.6.1 pero con los datos de las competiciones.

4.6.7 Guardar grupos (requisito cumplido: 13)

Ídem que el apartado 4.6.1 pero con los datos de los grupos de la aplicación.

4.6.8 Guardar respuestas juego (requisito cumplido: 53)

Ídem que el apartado 4.6.1 pero con las respuestas que se han dado en la fase de juego.

4.7 CASO DE ESTUDIO

Como se ha mencionado anteriormente, al momento del desarrollo de esta memoria, se ha lanzado una experiencia usando esta aplicación en una asignatura de grado. La asignatura de grado en concreto es Arquitectura de Sistemas del grado de Ing. Tecnologías de Telecomunicación de segundo curso primer cuatrimestre. Los contenidos de la asignatura incluyen conceptos de Sistemas Operativos, concurrencia y sincronización, aprendizaje del lenguaje C, creación de procesos, hilos y comunicación entre los mismos.

En la asignatura, en este curso 15-16, hay matriculados 118 alumnos, divididos entre el grupo bilingüe con 33 alumnos y el grupo en castellano con 85. Este grupo magistral se divide en dos grupos reducidos, el grupo 91 con 41 alumnos y el 92 con 44.

Se ha decidido lanzar la experiencia de forma escalonada para los tres grupos de la asignatura:

- El grupo 91 realiza los temas sobre creación de hilos y cerrojos.
- El grupo 92: realiza los temas sobre semáforos.
- El grupo 95: realiza los temas sobre monitores.

Por otra parte, se ha decidido ofrecer a los alumnos la posibilidad de obtener 0,25 puntos opcionales sobre la nota final: 0.1 puntos si completan todas las fases y 0.15 a criterio de la profesora que imparte la materia, dependiendo del desempeño de cada alumno.

Las fechas de las fases para los grupos son las siguientes:

- Fase escritor. Grupo 91 del 9-9 hasta el 22-9 (inclusive). Grupo 92 del 23-9 hasta 6-10 (inclusive). Grupo 95 del 28-9 hasta el 11-10 (inclusive).
- Fase revisor. Grupo 91 del 23-9 hasta el 27-9 (inclusive). Grupo 92 del 7-10 hasta el 11-10 (inclusive). Grupo 95 del 12-10 hasta el 16-10 (inclusive).
- Fase corrector. Grupo 91 del 28-9 hasta el 1-10 (inclusive). Grupo 92 del 12-10 hasta el 15-10 (inclusive). Grupo 95 del 17-10 hasta el 20-10 (inclusive).
- Fase corrección profesor. Grupo 91 del 2-10 hasta el 5-10 (inclusive). Grupo 92 del 16-10 hasta el 19-10 (inclusive). Grupo 95 del 21-10 hasta el 24-10 (inclusive).
- Fase juego. Grupo 91 del 6-10 hasta el 18-10 (inclusive). Grupo 92 del 20-10 hasta el 2-11 (inclusive). Grupo 95 del 25-10 hasta el 7-11 (inclusive).

Como se puede ver, en el momento de la escritura de esta memoria, el grupo 91 ha finalizado la fase escritor y ha empezado ya con la fase revisor, mientras que el grupo 92 y 95 ha empezado con la fase escritor.

En este grupo 91, de 41 alumnos han realizado la primera fase 36, lo que se puede apreciar como una buena recepción. Una primera evaluación de las preguntas iniciales, realizadas, hace pensar que están bien pensadas y estructuradas. Por otro lado, en la fase revisor se hay solo 3 alumnos, por lo que todavía no se proporcionan datos para ofrecer una conclusión objetiva.

Para poder evaluar la ganancia de aprendizaje de los alumnos, la profesora realizó una prueba diagnóstica el primer día de clase, evaluando el nivel de conocimientos de los alumnos. Asimismo, los grupos no hacen las preguntas de la misma parte del temario, para así poder evaluar si el realizar la competición ha mejorado sus conocimientos en la parte del temario que les ha tocado.

Además, para el grupo 91 no se ofrece la posibilidad de realizar revisiones opcionales, mientras que para el grupo 92 y 95 sí, con el fin de comprobar si el hecho de realizar revisiones opcionales mejora el aprendizaje.

Se pretende volver a lanzar la experiencia más adelante en la asignatura para comprobar sus efectos en el aprendizaje de un lenguaje.

Por último, se pretende analizar todos los datos con vista a realizar una publicación de los mismos en una conferencia o revista.

4.8 RESUMEN

Una vez finalizado el capítulo, se obtienen resultados favorables sobre el funcionamiento de la aplicación. Con esto, se puede observar que los requisitos que se establecieron en el capítulo 3 quedan demostrados y que el funcionamiento de la aplicación es correcto a través de las capturas de pantallas que se han añadido para poder verificar las funcionalidades.

Capítulo 5

CONCLUSIONES Y TRABAJO FUTURO

En este capítulo se va a explicar las conclusiones que se han obtenido a la hora de realizar el desarrollo de la aplicación, y los posibles trabajos futuros que se pueden implementar para poder mejorarlo.

5.1 CONCLUSIONES

Este trabajo fin de grado ha demostrado cómo se puede realizar una aplicación Cliente-Servidor basado en el modelo MVC.

Se ha creado una aplicación web dinámica en el que el principal objetivo es realizar un sistema de competición y revisión entre pares, para que los alumnos puedan aprender de los contenidos de la asignatura de una forma más divertida y así levantar el interés por parte de ellos.

Con este trabajo, se han podido estudiar varias tecnologías disponibles para el desarrollo de aplicaciones web y se ha podido escoger aquellas que tienen un mejor rendimiento y facilidad a la hora de usarlas para crear el funcionamiento requerido.

En primer lugar, se ha obtenido un conocimiento más amplio sobre el desarrollo de una aplicación web basada en un modelo MVC. Se ha programado en JAVA, ya que como se vio en el “estado del arte”, tiene más ventajas frente a las demás descritas. MySQL como base de datos, como bien se pudo apreciar en las tablas 2.1 y 2.2, tiene mejores características que las demás, a excepción de Oracle que por su alto coste no se ha podido utilizar para el desarrollo.

En segundo lugar, sobre la competición cabe destacar que esta herramienta va a servir de mucha ayuda para aquellos alumnos que no se sientan muy motivados en la realización de la asignatura. Normalmente, suelen haber estudiante que no les guste estudiar la materia como tal, pero si se les hace a través de un juego, como en este caso, pueden mostrar más interés ya que pueden mejorar su evaluación y aprender más.

Por último, cabe destacar el aprendizaje que se ha obtenido al momento de realizar esta aplicación. No solo ha servido para afianzar conocimientos en ámbitos donde no se tenía mucha experiencia, sino que también ha servido para obtener otros nuevos y agilizar el proceso de desarrollo de una aplicación.

5.2 TRABAJO FUTURO

Este trabajo fin de grado está siendo evaluado en una asignatura de la universidad Carlos III en una asignatura de grado tal y como se explicó en el apartado 4.8. Dentro del trabajo a corto plazo se encuentran los siguientes ítems:

- Terminar la experiencia
- Realizar un test final para medir la ganancia de aprendizaje (este coincidirá con un examen sumativo de evaluación continua de la asignatura).
- Analizar los datos de la experiencia
- Redactar un artículo para conferencia

Además, dependiendo de los resultados de esta primera experiencia, se pretende volver a lanzarla más adelante en la asignatura, con el fin de comprobar si este método es efectivo en el aprendizaje de los conceptos básicos teóricos de un lenguaje de programación y así, con las dos experiencias, poder publicar esto en un artículo de revista.

Si los resultados son los esperados, esta experiencia se repetirá en la asignatura en ediciones futuras.

Ya fijándonos en el lado más técnico y de mejora de las funcionalidades ofrecidas por la aplicación, algunas posibles mejoras para este trabajo son:

- Listar preguntas progresivamente
Una posible mejora sería que el profesor accediera a las preguntas directamente sin necesidad de volver a la lista para que sea un proceso más rápido.
- Mejora de la apariencia
La información que se muestra en la aplicación está ordenada y clara, pero no cuenta con un efecto visual muy llamativo. Al haberse centrado más en el desarrollo de la funcionalidad, se han creado páginas básicas para poder mostrar la información requerida. Para mejorar esto, sería necesario crear archivos CSS en los que el estilo de las páginas sea más fresco y llame más la atención.
- Listar preguntas para revisiones
La manera en la que se ha implementado la funcionalidad de listar las preguntas en la fase revisor, se basa en mostrar aquellas preguntas que no hayan sido revisadas. En caso de que todas tengan revisión, se escogerán aleatoriamente. Una mejora podría ser listar las preguntas basándose en el número de revisiones que tengan, es decir, se listarían primero aquellas que tienen menos revisiones y en caso de tener igual número de revisiones, se escogerían aleatoriamente.
- Estadísticas competición
La competición presenta la posibilidad de consultar los resultados de la fase juego para tener conocimiento del número de alumnos que han acertado y fallado las preguntas. Esta mejora podría realizarse añadiendo gráficas en las que se comparen las preguntas que se han jugado con el número de alumnos que las han contestado bien y los que han fallado para tener un mejor efecto visual de cómo han ido mejorando o empeorando los alumnos.
- Resultados de alumnos

El profesor es capaz de ver el resultado del alumno que seleccione. La mejora que se podría implementar aquí sería la opción de representar gráficamente los umbrales de puntuaciones con el número de alumnos que están en ese umbral de notas. De esta forma el profesor tendría una idea más clara y general de que notas son las más destacadas.

- Consultar fechas de realización de fase
El profesor es capaz de consultar las fases que ha hecho cada alumno. Una mejora podría ser, consultar también la fecha en la que ha realizado la fase. Esto serviría para saber quién ha realizado la fase al inicio o quién lo dejó para el final. De este modo podría evaluar el interés presentado por los alumnos
- Consultar alumnos preguntas falladas
El profesor es capaz de consultar las preguntas que han fallado los alumnos. Una mejora sería, que el propio alumno fuese capaz de ver quienes han fallado o acertado las preguntas que él ha creado. Del mismo modo también podría consultar el número de fallos y aciertos que han tenido estas.
- Ranking de alumnos
Los alumnos podrían ser capaces de ver el ranking de la competición. Podrían consultar los alumnos y los aciertos que han tenido de las preguntas que se han jugado y saber si están entre los primeros puestos o entre los últimos. De esta manera, lo que se pretende conseguir es que los alumnos se motiven para llegar al primer puesto.
- Aplicación en inglés
La aplicación se ha desarrollado en idioma español. Esto puede resultar difícil para aquellos alumnos que estén de intercambio y estén cursando la asignatura. Por lo tanto, como mejora podría introducirse la opción de escoger el idioma, español o inglés, en el que se desee abrir la aplicación.
- Ver resultados obtenidos
El alumno es capaz de ver el resultado de la fase realizadas pudiendo ver la puntuación que le han dado el profesor y los alumnos. Una mejora sería el poder consultar todas las notas dadas al alumno individualmente. Así, el alumno podría saber que preguntas han sido mejor o peor calificadas, y si los alumnos y el profesor piensan que ha realizado las revisiones correctamente o no.

Capítulo 6

PLANIFICACIÓN Y PRESUPUESTO

En este apartado se va a describir las diferentes etapas que ha seguido el presente trabajo. Son una serie de pasos que se han realizado para poder asegurar el cumplimiento correcto y las necesidades de la aplicación. Esto es lo que se le llama planificación.

Por otro lado, también se detallará el presupuesto final del trabajo como resultado de las etapas que se han seguido y del material usado para el desarrollo del trabajo.

6.1 PLANIFICACIÓN

El trabajo fin de grado ha tenido una duración de dos meses aproximadamente. Para empezar a realizar dicho trabajo, primero se tuvo que hacer un estudio de las posibles tecnologías que hay y cuales se adaptarían mejor a las características.

En la tabla 6.1 se presentan las diferentes tareas que se han llevado a cabo con el tiempo empleado en cada una de ellas.

	TAREA	FECHA INICIO	FECHA FINAL
1	Estudio de las diferentes tecnologías para el desarrollo de la aplicación web.	1/07/2015	10/07/2015
2	Repaso de java, jsp y modelo MVC.	10/07/2015	17/07/2015
3	Instalación de la máquina virtual Ubunuru, apache tomcat y MySQL	17/07/2015	19/07/2015
4	Creación del diseño, modelos, base de datos, vistas.	20/07/2015	27/07/2015
5	Implementación del código para la realización de la aplicación.	27/07/2015	30/08/2015
6	Corregir errores	28/08/2015	1/09/2015
7	Añadir mejoras	1/09/2015	3/09/2015
8	Pruebas de la aplicación	4/09/2015	6/09/2015
9	Instalación en el servidor de la universidad	7/09/2015	8/09/2015

Tabla 6.1 Planificación dividida en tareas

Como se puede comprobar, la duración del trabajo ha sido algo más de dos meses. A continuación se van a explicar de forma breve las tareas de la tabla:

- Tarea 1: lo primero que se debe hacer al iniciar el proyecto, es estudiar el funcionamiento de las aplicaciones web y de las tecnologías que se pueden emplear para su realización.
- Tarea 2: después de saber las tecnologías que se pueden emplear, se tendrá que afianzar los conocimientos para utilizar dichas tecnologías. En este caso se tendrá que dar un repaso a lo estudiado y además, adquirir conocimientos nuevos para poder realizar mejor el trabajo.
- Tarea 3: una vez que se hayan adquirido los conocimientos necesarios para poder empezar con la aplicación, se deberá instalar el entorno donde se va a trabajar. En este caso se decidió usar Ubuntu para realizar la aplicación, por lo que se tuvo que instalar la máquina virtual de Ubuntu. Además, hay que instalar el servidor donde se va a lanzar la aplicación, en este caso Apache Tomcat, e instalar la base de datos MySQL. Para la instalación de la máquina virtual, servidor y base de datos, se ha hecho uso de tutoriales [24][25].
- Tarea 4: cuando se tenga todo el entorno de trabajo instalado, se van a establecer los modelos que van a seguir los datos de la aplicación. Se tendrá que determinar la estructura que va a tener la aplicación, los datos que se van a guardar, los datos que se van a devolver, el modelo de las páginas web que va a tener, el diseño de la aplicación, etc. Esta es la base de la realización del trabajo, por lo que se necesita aclarar bien los modelos a seguir para poder implementar el código.
- Tarea 5: una vez que se tenga toda la estructura y toda la información necesaria sobre qué es lo que se necesita de la aplicación, se procederá a la implementación del código para que la aplicación funcione de la forma deseada.
- Tarea 6: después de haber creado el código de la aplicación, se deberán corregir posibles errores de programación o errores mínimos que haya, para dejar todo depurado y poder implementar más funcionalidades a parte de las básicas.
- Tarea 7: cuando se hayan corregido los errores, se podrán introducir mejoras a la aplicación. Estas mejoras incluyen mejoras en el código, en la forma de presentar la información, la lógica de la aplicación, nueva funcionalidad, etc.
- Tarea 8: antes de dar por finalizado el trabajo, se deberán hacer pruebas para comprobar el correcto funcionamiento del proyecto. En dichas pruebas, se completará el proceso que se requiere en la aplicación. Si el proceso logra completarse correctamente, entonces la aplicación ya podrá usarse. En caso de que no logre completarse, se deberán corregir los errores que hayan.
- Tarea 9: una vez se han hecho las pruebas necesarias y se tiene confirmación de que cumple con los requisitos propuestos, se deberá instalar en el servidor del sitio donde se quiera lanzar dicha aplicación, en este caso se trata de instalarlo en los servidores de la universidad.

La realización de las tareas, ha seguido una forma secuencial aunque ha habido situaciones en las que se ha tenido que volver a alguna tarea anterior para modificarla.

La figura 6.1 muestra el diagrama de Gantt con cada una de las tareas que se han descrito anteriormente:

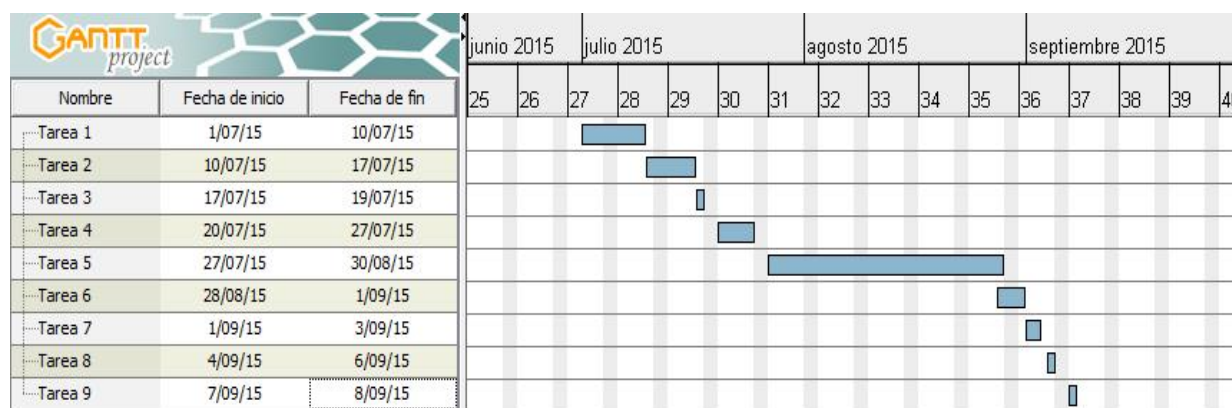


Figura 6.1 Diagrama de Gantt de la planificación

6.2 PRESUPUESTO

En este apartado se va a presentar el presupuesto total para realizar la aplicación, que se va a calcular teniendo en cuenta dos datos:

- **Horas dedicadas:** serán las horas dedicadas a realizar las tareas que se describieron en el apartado anterior para el desarrollo del trabajo. La tabla 6.2 muestra las horas empleadas en cada tarea.

Tarea	Horas
1	60
2	65
3	30
4	70
5	330
6	40
7	24
8	30
9	1

Tabla 6.2 Horas empleadas

Estos valores se pueden agrupar de forma más general para tener una mejor visualización de las horas dedicadas a las partes más grandes del trabajo. La tabla 6.3 recoge dicha información:

FASE 1	DOCUMENTACIÓN E INSTALACIÓN	155 HORAS
FASE 2	DESARROLLO DE SW	495 HORAS
FASE 3	MEMORIA	140 HORAS

Tabla 6.3 Horas dedicadas agrupadas

A continuación, se va a describir la actividad desarrollada en cada fase:

- **Fase 1:** dentro de esta fase se encuentra el tiempo relacionado con el estudio de las tecnologías, el repaso de las empleadas, la instalación de estas y el aprendizaje
- **Fase 2:** en esta fase se incluye la implementación del código, así como las mejoras, el diseño de la aplicación, el modelado de datos y todo lo demás relacionado.
- **Fase 3:** es la fase que se emplea para la realizar la memoria del trabajo.

Con esta información, se puede calcular el tiempo total empleado entre las tres fases y en este caso las horas totales son 790.

El periodo en el que se ha realizado el trabajo fin de grado ha sido de dos meses, desde Julio hasta Septiembre.

- **Material empleado:** para la realización de este trabajo, se ha contado con el material que se presenta a continuación en la tabla 6.4.

Objeto	Uso	Coste
Ordenador	Desarrollo software	800 euros

Tabla 6.4 Material empleado

Al tratarse de una aplicación web, el único gasto que se ha realizado ha sido el del ordenador. A esto hay que añadirle el precio de las horas empleadas para el desarrollo de la aplicación. Se establece que el precio por hora de dedicación es de 25 euros sin incluir el IVA. Por lo tanto, sabiendo esto se podrá calcular el presupuesto final, representado en la tabla 6.5:

GASTOS	PRECIO SIN IVA
800 + 495 (horas desarrollo SW) * 25 (euros)	13175 euros

Tabla 6.5 Presupuesto final

APÉNDICE A

A.1 VALIDACIÓN DE REQUISITOS

A.1.1 INTRODUCCIÓN

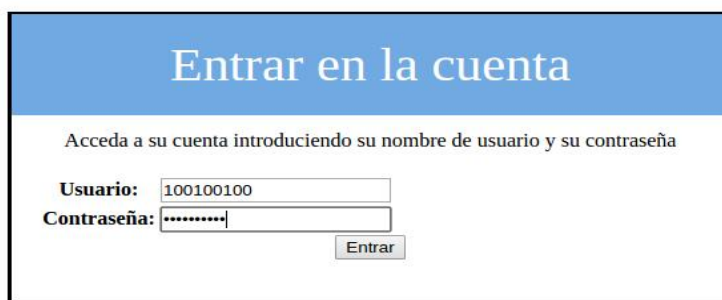
Este capítulo complementa al capítulo 4 de validación y requisitos, mostrando las diferentes capturas de pantalla que se obtienen de las pruebas realizadas en la aplicación.

A.1.2. ACCESO A LA APLICACIÓN

A.1.2.1 Acceder a la aplicación

Se mostrarán las diferentes capturas de pantalla para el acceso a la aplicación.

- Acceso alumno.



Entrar en la cuenta

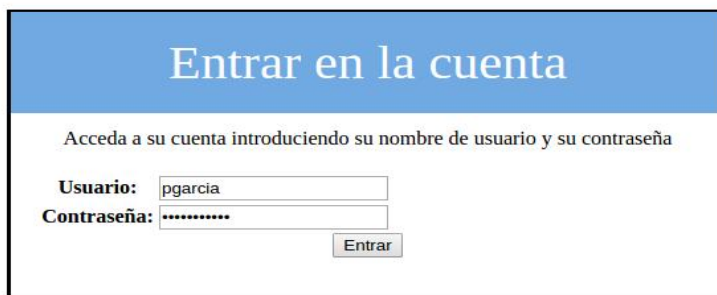
Acceda a su cuenta introduciendo su nombre de usuario y su contraseña

Usuario:

Contraseña:

Figura A.1 Acceso alumno

- Acceso profesor.



Entrar en la cuenta

Acceda a su cuenta introduciendo su nombre de usuario y su contraseña

Usuario:

Contraseña:

Figura A.2 Acceso profesor

Entrar en la cuenta

Acceda a su cuenta introduciendo su nombre de usuario y su contraseña

La contraseña no es correcta

Usuario:

Contraseña:

Figura A.3 Contraseña incorrecta

Entrar en la cuenta

Acceda a su cuenta introduciendo su nombre de usuario y su contraseña

Login no válido

Usuario:

Contraseña:

Figura A.4 Login no válido

A.1.2.2 Cambio de contraseña

Longitud mínima de la contraseña es de 6 caracteres. Debe contener números y letras.

Contraseña actual:

Nueva contraseña:

Repita contraseña:

Figura A.5 Cambio de contraseña



Figura A.6 Contraseña cambiada

En caso de no introducir correctamente la contraseña actual o que la contraseña nueva no cumpla con las condiciones puestas:

La contraseña actual que ha introducido no es correcta

Figura A.7 Contraseña actual incorrecta

Contraseña actual:

La contraseña debe tener números y letras

Nueva contraseña:

Repita contraseña:

Figura A.8 Contraseña no cumple condiciones

A.1.3 GESTIONAR GRUPOS

Se mostrarán las diferentes capturas de pantalla relacionadas con la gestión de los grupos.

A.1.3.1 Consultar alumnos de un grupo

[Ver grupos](#) | [Editar grupos](#) | [Crear grupos](#) | [Borrar grupo](#)

Figura A.9 Opciones gestión grupo profesor administrador

[Ver grupos](#) | [Editar grupos](#)

Figura A.10 Opciones gestión grupo profesor no administrador

En caso de que se quiera consultar un grupo vacío:

No hay alumnos en este grupo

Figura A.11 No alumnos grupo

A.1.3.2 Editar grupos

[Añadir alumnos a partir de csv](#) | [Añadir alumnos de forma manual](#) | [Modificar alumnos](#)

Figura A.12 Opciones editar grupos

A.1.3.2.1 Editar alumno de un grupo

Lista de alumnos del grupo 91

NIA	Nombre	Apellidos
100100100	Antonio	Gómez Pérez

Figura A.13 Lista alumnos a editar

Datos del alumno

NIA	Nombre	Apellidos	Grupo
100100100	Antonio	Gómez Pérez	91

Figura A.14 Opciones a realizar

A.1.3.3 Crear grupo

[Crear grupo vacío](#) | [Crear grupo importando alumnos](#)

Figura A.15 Opciones crear grupo

A.1.3.3.1 Crear grupo vacío

El grupo ya existe

Figura A.16 Grupo existente

A.1.3.3.2 Crear grupo importando alumnos

**Añadir alumnos importando
archivo**

INTRODUCIR ARCHIVO PARA CARGAR ALUMNOS

Ningún archivo seleccionado

Figura A.17 Importar archivo CSV

Validar la creación del nuevo grupo

¿Son correctos estos datos?

NIA	Login	Nombre	Apellidos
100282983	100282983	Grecia	Vera Villajulca
100285003	100285003	Alexandre	Solves Almeida
123456789	123456789	Pedro	Gutierre Perlin
111222333	111222333	Ivan	Rodriguez Pereda
111111111	111111111	Sandra	Garcia García

Grupo:

Crear

No crear

Figura A.18 Listar alumnos CSV y crear grupo

A.1.4 GESTIONAR PROFESORES

[Añadir profesor](#) | [Editar profesor](#) | [Ver profesores](#)

Figura A.19 Opciones gestión profesor

A.1.4.1 Añadir profesor

Añadir a un profesor

Introducir datos del profesor:

Login:

pgarcia

Contraseña:

profesor123

Nombre:

Pedro

Apellidos:

García García

Grupo:

91 ▾

Administrador:

No ▾

Si

No

Limpiar

Crear

Figura A.20 Registro profesor

A.1.4.2 Editar profesor

Login	Nombre	Apellidos	Grupo
pgarcia	Pedro	García García	91

Login:

pgarcia

Nombre:

Apellidos:

Grupo:

91 ▾

Limpiar

Guardar

Figura A.21 Editar profesor

A.1.4.3 Ver profesores

Lista de profesores del grupo
96

Login	Nombre	Apellidos
lsanchez	Luis	Sanchez García

Figura A.22 Ver profesores grupo

No hay profesores en este grupo

Volver

Figura A.23 No profesores grupo

A.1.5 GESTIONAR COMPETICIONES

[Todas las competiciones](#) | [Competiciones abiertas](#) | [Competiciones cerradas](#) | [Corregir preguntas finales de alumnos](#) | [Seguimiento de alumnos](#) | [Estadísticas competicion](#) | [Crear competicion](#) | [Borrar competicion](#)

Figura A.24 Gestionar competiciones

A.1.5.1 Listar todas las competiciones

Seleccione un grupo

Grupo: 91 ▼

Listar competiciones

Figura A.25 Seleccionar grupo para listar competiciones

Lista de las competiciones
del grupo 91



Fecha inicio FASE ESCRITOR	Preguntas FASE ESCRITOR	Fecha inicio FASE REVISOR	Revisiones FASE REVISOR	Fecha inicio FASE CORRECTOR	Correcciones FASE CORRECTOR	Fecha inicio CORRECCION PROFESOR	Fecha inicio JUEGO	Preguntas a jugar JUEGO	Fecha fin JUEGO
2015-09-15	2	2015-09-16	1	2015-09-17	2	2015-09-18	2015-09-19	2	2015-09-20
2015-09-15	2	2015-09-15	2	2015-09-15	2	2015-09-15	2015-09-15	3	2015-09-15

Figura A.26 Lista de todas las competiciones del grupo 91

A.1.5.2 Listar competencias cerradas

No hay competencias cerradas

[Volver](#)

Figura A.27 No hay competencias cerradas

A.1.5.3 Corregir preguntas finales alumno

Podrá empezar a corregir las preguntas y revisiones de esta competición el 2015-09-17 y finalizar antes del 2015-09-17.

[Corregir](#)

Figura A.28 Acceso fase corrección profesor

NIA	Nombre	Apellidos	Preguntas hechas
100285003	Alexandre	Solves Almeida	2/2
123456789	Pedro	Gutierre Perlin	2/2

[Validar](#)

Figura A.29 Lista de alumnos

Pregunta fase Escritor	Corregida	Pregunta fase Corrector	Corregida Profesor	Pregunta final profesor
Enunciado pregunta 1 alumno2 para fase escritor	SI	pregunta 1 cambiada alumno2 fase corrector	NO	pregunta 1 cambiada alumno2 fase corrector
Enunciado pregunta 2 alumno 2 para fase escritor	NO	Enunciado pregunta 2 alumno 2 para fase escritor	NO	Enunciado pregunta 2 alumno 2 para fase escritor

Figura A.30 Lista preguntas alumno

Esta competición no dispone de alumnos que hayan hecho la fase Corrector

[Volver](#)

Figura A.31 No alumnos fase corrector

PREGUNTA INICIAL FASE ESCRITOR

Enunciado pregunta: Enunciado pregunta 1 alumno2 para fase escritor

Opción 1: respuesta 1 alumno2

Opción 2: respuesta 2 alumno2

Opción 3: respuesta 3 alumno2

Opción 4: respuesta 4 alumno2

Opción correcta: respuesta 2 alumno2

Puntuación a la pregunta

Figura A.32 Pregunta de la fase escritor

PREGUNTA CAMBIADA EN LA FASE CORRECTOR

Enunciado pregunta:

pregunta 1 cambiada alumno2 fase corrector

Opción 1:

respuesta 1 alumno2

Opción 2:

respuesta 2 alumno2

Opción 3:

respuesta 3 alumno2

Opción 4:

respuesta 4 alumno2

Opción Correcta:

respuesta 3 alumno2

Puntuación a la pregunta

Figura A.33 Pregunta de la fase corrector

Pregunta fase Escritor	Corregida	Pregunta fase Corrector	Corregida Profesor	Pregunta final profesor
Enunciado pregunta 1 alumno2 para fase escritor	SI	pregunta 1 cambiada alumno2 fase corrector	NO	pregunta 1 cambiada alumno2 fase corrector
Enunciado pregunta 2 alumno 2 para fase escritor	NO	Enunciado pregunta 2 alumno 2 para fase escritor	SI	pregunta 2 alumno 2 cambiada profesor

Figura A.34 Lista de preguntas con corrección profesor

Las puntuaciones de la pregunta y revisiones se han guardado

Volver

Figura A.35 Puntuaciones guardadas

Las preguntas de estos alumnos aun no han sido validadas para la fase juego

Volver

NIA	Nombre	Apellido
123456789	Pedro	Gutierre Perlin

Figura A.36 Alumno no preguntas validadas

El juego se encuentra en la fase corrector
del alumno

Volver

Figura A.37 Fin fase corrección profesor

A.1.5.4 Jugar competición

A.1.5.4.1 Fase escritor

Lista de competiciones										
Fecha inicio FASE ESCRITOR	Preguntas FASE ESCRITOR	Fecha inicio FASE REVISOR	Revisiones FASE REVISOR	Fecha inicio FASE CORRECTOR	Correcciones FASE CORRECTOR	Fecha inicio CORRECCION PROFESOR	Fecha inicio JUEGO	Preguntas a jugar JUEGO	Fecha fin JUEGO	
2015-09-15	2	2015-09-16	1	2015-09-17	2	2015-09-18	2015-09-19	2	2015-09-20	Seleccionar
2015-09-17	2	2015-09-17	2	2015-09-18	2	2015-09-18	2015-09-18	2	2015-09-18	Seleccionar

Figura A.38 Lista de competiciones a jugar

Aun no puede acceder a la competición

Volver

Figura A.39 Mensaje acceso competición

Competición

La competición empezará el 2015-09-17

Competir

Figura A.40 Acceso a la fase escritor

Validar preguntas de la fase de ESCRITOR

¿Son correctas estas preguntas?

Enunciado pregunta:

¿Enunciado de pregunta para el juego inicial?

Opción 1:

Aquí se escribe la opción 1

Opción 2:

Aquí se escribe la opción 2

Opción 3:

Aquí se escribe la opción 3

Opción 4:

Aquí se escribe la opción 4

Opción correcta:

Opcion 3 ▼

Figura A.41 Validación de preguntas

A.1.5.4.2 Fase revisor

La fase de REVISOR empezará el 2015-09-17

Entrar

Figura A.42 Acceso fase revisor

A.1.5.4.3 Fase corrector

La fase de CORRECTOR empezará el 2015-09-17

Entrar

Figura A.43 Acceso fase corrector

Corregir pregunta

Enunciado pregunta:

Opcion 1:

Opcion 2:

Opcion 3:

Opcion 4:

Opcion correcta:

Opcion 1 ▼

Limpiar Crear

Figura A.44 Corregir pregunta

Su pregunta ha sido cambiada

Volver

Figura A.45 Pregunta cambiada

Se ha validado la corrección

Volver

Figura A.46 Corrección validada

A.1.5.4.4 Fase juego

El JUEGO empezará el 2015-09-17

Entrar

Figura A.47 Acceso fase juego

El JUEGO finaliza el 2015-09-18

Volver

Figura A.48 Final de competición

A.1.5.5 Seguimiento de alumnos

[Alumnos que han creado preguntas](#) | [Alumnos que han hecho revisiones](#) | [Alumnos que han hecho correcciones](#) | [Alumnos que han jugado](#) | [Fase que ha realizado cada alumno](#) | [Resultados de alumnos](#)

Figura A.49 Opciones de seguimiento

A.1.5.5.1 Alumnos que han creado preguntas

Pregunta inicial	Corregida	Pregunta final
Enunciado pregunta 1 alumno2 para fase escritor	NO	Enunciado pregunta 1 alumno2 para fase escritor
Enunciado pregunta 2 alumno 2 para fase escritor	NO	Enunciado pregunta 2 alumno 2 para fase escritor

Figura A.50 Lista preguntas iniciales y finales

Enunciado pregunta:

Enunciado pregunta 2 alumno 2 para fase escritor

Opción 1:

respuesta 1 alumno2

Opción 2:

respuesta 2 alumno2

Opción 3:

respuesta 3 alumno2

Opción 4:

respuesta 4 alumno2

Figura A.51 Pregunta detallada

A.1.5.5.2 Alumnos que han hecho revisiones

[Alumnos que han hecho revisiones obligatorias](#) | [Alumnos que han hecho revisiones opcionales](#)

Figura A.52 Opciones de revisiones

A.1.5.5.2.1 Alumnos que han hecho revisiones obligatorias

Enunciado pregunta	
¿Enunciado de pregunta para el juego inicial?	Seleccionar
¿Enunciado de pregunta 2 para el juego inicial?	Seleccionar

Figura A.53 Lista preguntas revisiones obligatorias

Revision y pregunta revisada

Enunciado pregunta:
 ¿Enunciado de pregunta para el juego inicial?

Opción 1:
 Aquí se escribe la opción 1

Opción 2:
 Aquí se escribe la opción 2

Opción 3:
 Aquí se escribe la opción 3

Opción 4:
 Aquí se escribe la opción 4

Opción correcta:
 Aquí se escribe la opción 3

Revisión
Puntuación de la pregunta: 6.7
Comentario:
 Comentario alumno2 para pregunta 1
Posible mejora:
 Posible mejora alumno2 para pregunta 2

COMENTARIO CORRECTOR:
 comentario alumno1 revision pregunta 1

Figura A.54 Pregunta y revisión asociada

No hay alumnos que hayan hecho revisiones

Volver

Figura A.55 No alumnos revisiones

A.1.5.5.2.2 Alumnos que han hecho revisiones opcionales

No hay alumnos que hayan hecho revisiones opcionales

Volver

Figura A.56 No alumnos revisiones opcionales

A.1.5.5.3 Alumnos que han hecho correcciones

No hay alumnos que hayan hecho correcciones

Volver

Figura A.57 No alumnos correcciones

A.1.5.5.4 Alumnos que han jugado

No hay alumnos que hayan jugado

Volver

Figura A.58 No alumno fase juego

A.1.5.5.5 Resultados de los alumnos por parte del profesor

Lista de preguntas falladas

Enunciado pregunta	
Enunciado pregunta 2 alumno 2 para fase escritor	Seleccionar
Enunciado pregunta 1 alumno2 para fase escritor	Seleccionar

Figura A.59 Preguntas falladas

A.1.5.5.6 Resultados del propio alumno

<u>Resultados como ESCRITOR</u>	<u>Resultados como REVISOR</u>	<u>Resultados como CORRECTOR</u>	<u>Fallos/Aciertos Juego Inicial</u>	<u>Fallos/Aciertos JUEGO</u>
---------------------------------	--------------------------------	----------------------------------	--------------------------------------	------------------------------

Figura A.60 Resultados alumno

A.1.5.5.6.1 Resultados como revisor

Fecha inicio REVISOR	Fecha fin REVISOR	Puntuación	Notas media profesor	Notas media alumnos
2015-09-17	2015-09-17	7.28	8.0	6.2

Figura A.61 Resultado fase revisor

A.1.5.5.6.2 Resultados como corrector

Fecha inicio CORRECTOR	Fecha fin CORRECTOR	Puntuación
2015-09-17	2015-09-17	9.0

Figura A.62 Puntuación fase corrector

No dispone de resultados

Volver

Figura A.63 No resultados fase corrector

A.1.5.5.6.3 Fallos y aciertos juego inicial

No dispone de resultados del juego de la fase revisor

[Volver](#)

Figura A.64 No resultados juego inicial

A.1.5.5.6.4 Fallos y acierto juego

No dispone de resultados de la fase juego

[Volver](#)

Figura A.65 No resultados juego final

A.1.5.6 Estadísticas competición

[Estadística detallada competición](#) | [Estadística para exportar a csv](#)

Figura A.66 Opciones de estadísticas

A.1.5.6.1 Estadística detallada competición

Lista de alumnos que han acertado la pregunta

NIA	Nombre	Apellidos
123456789	Pedro	Gutierre Perlin

Figura A.67 Alumno que ha acertado la pregunta

A.1.5.6.2 Estadísticas para exportar en CSV

Enunciado pregunta	Numero personas acertadas	Numero personas falladas
¿Enunciado de pregunta para el juego inicial?	0	0
pregunta 1 cambiada alumno2 fase corrector	0	1
pregunta 2 alumno 2 cambiada profesor	1	0
pregunta 2 alumno1 cambiada profesor	0	0

[Exportar a csv](#)

EstadísticaCsv.csv

[Mostrar todas las descargas...](#)

Figura A.68 Descarga de las preguntas en CSV

A.1.5.7 Crear competición

No puede haber una fecha posterior menor que una anterior.

[Volver](#)

Figura A.69 Error fechas competición

A.1.5.8 Borrar competición

Lista de las competiciones
del grupo 91



Fecha inicio FASE ESCRITOR	Preguntas FASE ESCRITOR	Fecha inicio FASE REVISOR	Revisiones FASE REVISOR	Fecha inicio FASE CORRECTOR	Correcciones FASE CORRECTOR	Fecha inicio CORRECCION PROFESOR	Fecha inicio JUEGO	Preguntas a jugar JUEGO	Fecha fin JUEGO	
2015-09-15	2	2015-09-16	1	2015-09-17	2	2015-09-18	2015-09-19	2	2015-09-20	Borrar
2015-09-15	2	2015-09-15	2	2015-09-15	2	2015-09-15	2015-09-15	3	2015-09-15	Borrar

Figura A.70 Borrar competición deseada

A.1.6 GUARDAR DATOS

[Guardar preguntas](#) | [Guardar revisiones](#) | [Guardar correcciones](#) | [Guardar alumnos](#) | [Guardar profesores](#) | [Guardar competiciones](#) | [Guardar grupos](#) | [Guardar respuestas juego](#)

Figura A.71 Opciones para guardar datos

A.1.6.1 Guardar preguntas

Todos los datos de las competiciones se han guardado correctamente

[Volver](#)

Figura A.72 Datos guardados correctamente

BIBLIOGRAFÍA

- [1] Declaración de Bolonia, “El espacio europeo de la enseñanza superior,” Declaración conjunta de los ministros europeos de educación reunidos en Bolonia el 19 de junio de 1999.
- [2] Raquel M. Crespo-García, Iria Estévez-Ayres, Carlos Delgado Kloos. “Evaluación para el aprendizaje aplicación revisión entre iguales: caso de estudio en una asignatura de programación”. Aceptado en Jornadas de Innovación Educativa, 2015 (JIE 2015) Palma de Mallorca, España, Octubre 2015.
- [3] Ediger, Marlow. “Cooperative Learning versus Competition: Which Is Better?”, Diciembre 2001.
- [4] Jefatura del Estado «BOE» núm. 298, de 14 de diciembre de 1999. “Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal”. Última modificación 5 de Marzo 2011.
- [5] Agencia Española de Protección de Datos, “Guía de seguridad de datos”, Editorial: NILO Industria Gráfica, S.A. Consultado: 3-7-2015
- [7] K. Topping, “Peer Assessment Between Students in Colleges and Universities”, *Review Of Educational Research*, vol. 68, no 3, 1998.
- [8] Raquel M. Crespo García, Julio Villena Román. “Revisión entre pares como instrumento de aprendizaje. Una experiencia práctica.”, Marzo 2005.
- [9] Ewald Geschwinde and Hans-Jürgen Schönig, “PostgreSQL Developer's Handbook”, Abril 2002.
- [10] Jérôme Gabillaud, “Oracle 11g SQL, PL/SQL, SQL*Plus”, Enero 2012.
- [11] Robin Nixon, "Learning PHP, MySQL, Javascript, CSS, HTML5. A step by step guide to creating dynamic web sites". 3ª Edición, Editorial O' Reilly, Mayo 2012.
- [12] Miquel Colobran Huguet, Josep Maria Arqués Soldevilla, Eduard Marco Galindo, “Administración de sistemas operativos en red”, Primera edición, Noviembre 2008.
- [13] Aleksa Vukotic and James Goodwill, “Apache Tomcat 7”, 2011.
- [14] Mohammed J. Kabir, “Servidor Apache 2”, Editorial Anaya Multimedia, 2003.
- [15] Thierry Groussard, “Java 7: Los fundamentos del lenguaje Java”, Editions ENI, Marzo 2012.
- [16] John Zelle, “Python programming: an introduction to computer science”, Franklin, Beedle & Associates Incorporated, 2004.
- [17] Olivier Heurtel, “PHP 5.5. Desarrollar un sitio web dinámico e interactivo”, Editions ENI, Febrero 2014.

- [18] Günter Born, “Compendium Html. Con XHTML, DHTML, CSS, XML, XSL y WML”, 2000.
- [19] Ralph G. Schulz, “Diseño web con CSS”, Primera edición, Enero 2009.
- [20] W3School. Css. URL: <http://www.w3schools.com/css/> Consultado: 13-7-2015
- [21] Eric Freeman, Elisabeth Robson, “Head First JavaScript Programming”, 1º Edition, Marzo 2014
- [22] Juan Pavón Mestras, “El patrón Modelo-Vista-Controlador (MVC). Programación orientada a objetos facultad de informática”, Departamento de Ingeniería del Software e Inteligencia Artificial. Universidad Complutense de Madrid. 2008-2009.
- [23] Marco de Desarrollo de la Junta de Andalucía, “Patrón Modelo Vista Controlador”.
- [24] Olmos Miguelañez, Susana. "Evaluación formativa y sumativa de estudiantes universitarios: aplicación de las tecnologías a la evaluación educativa.". Tesis doctoral, Universidad de Salamanca (2008).